Spring 2009

A Method for Introducing Artificial Perception (AP) to Improve Human Behavior Representation (HBR) Using Agents in Synthetic Environments

Randall Bartholomew Garrett
Old Dominion University

A METHOD FOR INTRODUCING ARTIFICIAL PERCEPTION (AP) TO IMPROVE HUMAN BEHAVIOR REPRESENTATION (HBR) USING AGENTS IN SYNTHETIC ENVIRONMENTS

by

Randall Bartholomew Garrett
B. A. May 1976, University of Arkansas
M. S. May 1986, Marshall University

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirement for the Degree of

DOCTOR OF PHILOSOPHY

MODELING AND SIMULATION

OLD DOMINION UNIVERSITY
May 2009

Approved by:

Andreas Tolk (Director)

Ryland Gaskins III (Member)

John Sokolowski (Member)

Eric Weisel (Member)

UMI Number: 3356762

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

ABSTRACT

A METHOD FOR INTRODUCING ARTIFICIAL PERCEPTION (AP) TO IMPROVE HUMAN BEHAVIOR REPRESENTATION (HBR) USING AGENTS IN SYNTHETIC ENVIRONMENTS

Randall Bartholomew Garrett
Old Dominion University, 2009
Director: Dr. Roland Mielke

While psychology has shown that perception is very important for the human decision process, agent perception has not been covered in sufficient detail within the agent directed simulation field. To contribute to such a solution, an open challenge lies in capturing the knowledge of human sciences, such as psychology, and making this knowledge usable for engineers. This dissertation addresses perception by describing an experimental method where agent perception simulates human perception. In particular, it presents engineering methods based on accepted psychological approaches resulting in a proof of concept. To prove the feasibility, an Artificial Perception (AP) meta-model is presented using logical assumptions, generalized Validation and Verification (V&V) approaches, cognitive testing, and experimental comparisons with similar state-of-the-art agents.

Current publications show that Human Behavior Representation (HBR) in agent simulations remains a top priority for the modeling and simulation community. Agents are a promising way to represent human behavior. However, there are limited methods for measuring and validating perception within simulated environments. Within a simulated environment, there are no clearly established methods for understanding surrogate HBR perception relationships with their authentic human behavioral

representations. These present obstacles for advancing simulation of human behavior. The approach presented in this dissertation provides an engineering solution that allows bridging this gap and configuring agents based on human behavior evaluations using accepted psychological approaches, in particular, applied cognitive task analysis.

In many cases, the foundation for these engineering methods and the effects of agent perception are similar to "experimental frames" with some known potential for accomplishing a task. Effects of these tasks require accurate assessments beyond face validation. Agents play a critical role in HBR's capacity to accomplish specific tasks. An agent, in this sense, becomes less a matter of size or complexity, and more a matter of "viewpoint." This viewpoint makes it possible to infer an agent-oriented approach for conceptual modeling of AP, thus implying an agent-object relationship.

An experimental approach is provided using proven psychology perception measurements to help answer the question whether there is a correlation between human perception in the real world and its same representation in an artificial world. The results presented in this dissertation show that the method applied in the experimental approach can be generalized. Furthermore, the example used builds a proof of concept, as the resulting agent with AP reproduced the human behavior more closely than did the state-of-the-art approaches.

This is dedicated to all those who have given me
their unwavering love and support.

ACKNOWLEDGMENTS

I am grateful and indebted to my advisor and mentor Dr. Andreas Tolk for sharing his wisdom, support, and guidance. He has my highest honor and respect as an accomplished scientist and as an explorer of knowledge into this brave new world of artificial perception.

Furthermore, I am indebted to Dr. Eric Weisel, an expert in model semantic composability, for his guidance and support and to Dr. John Sokolowski, whose work with Recognition-Primed Decision making (RPD) and the human decision process, has served as an intellectual blueprint for this project.

I am also deeply grateful to Dr. Ryan Gaskins III, whose steadfast support has helped tremendously through this endeavor. Dr Gaskins' expert understanding of cognition and human experimentation has provided true inspiration.

Thank all of you deeply for serving on my Advisory Committee.

Most of all, I am grateful to my wife Pauline and family for their unwavering supporting throughout this endeavor.

I would like to sincerely thank all of my close friends within the modeling and simulation community for their friendship and encouragement.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

1.1. Thesis Statement

The field of psychology has proven models and explanations for human behavior that have been successfully relegated as roles for agents in Human Behavior Representation (HBR). An example of this is the use of perception in decision making. Agents are able to simulate human behavior and have been able to successfully accomplish behavioral tasks [Herrero et al. 2002]. For this to occur, supporting empirical data and confirming observations are required to ensure that the agents correctly address these tasks [DMSO 2003; Sloman 1996; Jack 1993]. In addition, agents representing a "viewpoint" infer conceptual modeling of perception through agent-object relationships [Riecken 1994; Minsky 1975]. Regarding decision making, the usage of a synthetically represented perception or methods to improve Artificial Perception (AP) are currently missing in Agent HBR. [1]

Goal-oriented behavior that is generated during perception recognizes whether a task of perceiving an object is complete. These perceptual responses, based on visual cues, require more introspection for agent models. In particular, visual cues are satisfactory for triggering recognition of these objects for automatic, reflex-like, spatial updating in humans [Riecke et al. 2005]. These triggers may be used as important drivers for HBR and provide insight as to how objects for agent behaviors may be perceived in a synthetic environment [Kahneman et al. 1992].

---

[1] This thesis uses the American Institute of Physics (AIP) style specification.

Mimicking human perception is not a new concept within the field of engineering; however, an effective method is needed to transfer perceptual knowledge from human sciences for applications in engineering. A central premise is that human perception is correlated to the same perception in an agent simulation. Selected psychological models help illuminate this. For example, Applied Cognitive Task Analysis (ACTA) methods allow observations of behavioral data that are gathered by extracting the working knowledge of Subject Matter Experts (SMEs) for a critical task "use case" [Freeman and Cohen 1998; Hoffman, *et al.* 1998; Thordson 1991]. The use of this psychological "thought model" draws a parallel between an agent's behaviors, represented by values from SME cognitive test scores, and compares them with values from a surrogate agent's behavior.

Quantifiable HBR cognitive responses relating to similar responses in an Agent-Based (AB) environment are measured. In part, these measurements assess the fidelity of the AB representation and the responses [Gross *et al.* 1999: Watson and McGaffey 2001]. A method for introducing AP within agent simulation that improves HBR calculates effectual cognitive test scores that crosswalk both the HBR for the task and the anticipated results of the agents performing the task. This method tests a significant measure for AP within a simulated HBR agent environment, thus allowing for comparisons to a human task correlate [Klein 1989, 1996].

It is possible to conduct experiments mimicking expert behavior based on visual cues. Using these cues, an effective approach applies validation for perception in an AB simulation of critical events. ACTA and critical decision methods allow for the creation of a cognitive representation for human perception in an agent environment. Using SME expert data from the cognitive experiments and implementing a method that introduces AP ensures accurate, simulated, HBR perception effects. This approach results in agent models that improve HBR and more accurately replicates the desired outcome of agent behavior "realism" for human perception [Garrett 2006]. The method includes, as proposed in this thesis:

1. Using logic, a dynamic mental model of perception, and generalized V&V techniques to develop an experiment
2. Eliciting a critical group task from Subject Matter Experts (SMEs)
3. Developing a scenario and dialogue based on SME perceptions for this task
4. Observing SME behavior by using gaming vignettes; collecting data, building visual *gestalts* [Wertheimer 1922]. and logically mapping these behaviors to agent representation
5. Executing an agent meta-model via critical scenarios developed from games to "trigger" agent behavior that mimics observed SME behavior
6. Experimenting, evaluating and comparing results to similar performances from state-of-the-art, rule-based agents that use perception.

Applied logic, ACTA, gaming scenarios and generalized V&V are components within the experimental method. Together, these seek a human expert correlate for perception that is mimicked by a similar AB task within HBR behavioral response. Also shown, is that reactive agents constructed using this method more closely resemble expert decision making from those similar behaviors exhibited by current state-of-the-art rule sets.

In summary, this thesis both documents a method of knowledge-transfer of human sciences for AB applications using proven psychological methods to extract expert behaviors and applies this to logical mapping in simulation. This logical mapping reproduces engineering methods and is rooted in established psychological models to create a representative meta-model.

1.2. Problem Domain

The underlying and documented research shows that proven psychological models for perception in decision making lack general procedures for transferring knowledge to a reproducible engineering method. This results in *ad-hoc* methods, yielding unreasonable approaches for replicating human knowledge as HBR. However, there are many proven examples of accurate perception measurements within the field of Psychology that have direct application to agent simulations. This transfer of human perception to a reproducible engineering method is required for producing realistic HBR agent behavior.

The problem domain specifically addresses transfering knowledge of human sciences to agent applications. This developed method uses agent behaviors that require the implementation of human perception. This ensures a realistic mapping of perception to the HBR agent simulation [Beer 2003; Cassimatis *et al.* 2004]. Recognized within the agent-directed community, agents with HBR are important and mapping of perception is essential to correctly simulate HBR agent responses [Fink *et al.* 2007; Simonin and Gechter 2006]. Regarding AP:

- Agents need perception for anticipatory responses
- Agents need meaning for what is perceived
- Agents need perception to build cognition
- Agents need to map perceived objects and events to a meta-model.

There are many established psychological methods for identifying perception elements, but a connection of these to engineering is needed. Implementing agents in a simulated environment requires the capability for them to perceive the environment and the ability to map this perception to an appropriate meta-model [Zeigler 2000].

Perception remains a key part of many proven psychological models and is recognized as having direct significance for expert decisions. The experimental premise is that if an agent HBR with AP closely mimics the actions of a human expert, this agent should reproduce expert decisions better than current state-of-the-art perception agents. [2]

Modeling perception is an area that is new to the agent-directed community and requires further exploration [Yilmaz and Ören, 2005]. Perception may generally be implied; however, there is no coherent HBR agent cognitive state remaining. This infers that an agent's expectation of the world and its anticipation of events require further definition. Without perception, agents perceived past and future realities that are not possible [Goldstein. 2002; Simonin and Gechter 2006; Wertheimer 1958].

A portion of the problem domain lies within the design for a best method of validating HBR AP. In this instance, V&V for HBR simulations and human perception

---

[2] State-of-the-art is used in this context as the current highest level of development and use for HBR perception.

involves a comparison of agent responses with anticipated real-life responses [Tanaka *et al.* 1996; Tanaka 2000]. In this case, HBR perception that is based on anticipation needs to correspond with the same perception representations found in an agent-artificial world [Butz 2007; Ören *et al.* 2004]. This includes defining methods for the following:

- Integration of multiple knowledge representations
- Validation of inference techniques
- Realistic integration of cognition, perception and action into the agent simulation.

When comparing this problem domain to agent models for HBR, it is evident that perception through "valid" responses based on the anticipation of a human's decision-making preference requires improvement [Cassimatis *et al.* 2004; Harvey *et al.* 2005]. In many cases, perceptions within these models are generally viewed at a top-down or macro-level within a much larger context, such as those approaches found within Situation Awareness (SA) where perception responses are viewed as an accumulation of object detection, sensor responses, or even as visual cues [Endsley 1996; Endsley *et al.* 2003; Firby 1987: Russel and Norvig 2003].

Examples are anticipatory systems that use predictions to determine an agent's behavior, (e.g., let future states affect present states). The basis of these are causal reasoning only and are limited [Dennis and Ahn, 2001]. They provide incomplete responses for the HBR of perception within their environmental relationships. Research into these areas does not yield usable instances of HBR perception experimentation or validation methodologies. Engaging psychological models that use cognitive experimental techniques, found within the field of SA, address this shortfall. This includes leveraging Situation Awareness Global Assessment Techniques (SAGAT) [Endsley, 1996]. and Situational Awareness Rating Techniques (SART) [Taylor 1989]. In addition, cognitive models referencing anecdotal information and mathematical test results compare actual perception using simulations in the HBR AP approach.

The literature shows that in many cases the development of rules for agents is performed on an *impromptu* basis, thus the builder, developer and implementer of the agent make many of the decisions. Accepted methods of psychology, which extract knowledge comprising of the cognitive elements, are not often found in current, agent-rule development approaches. Introducing methods provided in this dissertation seek to close this gap. This would provide a means for proven psychological techniques that measure cognitive elements of task oriented behavior to be used in building agents for HBR.

Transferring knowledge to HBR computer models encompasses both the behavior of a single human and the collective actions of a group [Yilmaz 2005, 2006]. In atomic form, HBR refers to a portrayal of humans; however, phenomena associated with HBR remains highly complex. A logical approach connecting observed human behavior specific to a problem set is needed to define the narrative of human actions for HBR. Limiting the scope of behavior to a particular problem set allows for an aggregation of these influences. For the HBR AP model, these human influences must logically map to those of the AB representations.

Within the problem domain, how to build an agent with perception is considered. An AP agent [Zeigler 2000]:

- Needs to perceive and recognize objects within the environment
- Needs a"meta-model" to represent the behavior
- Must be able to map perception to the "meta-model."

To successfully incorporate a perception method into agent design, human "projections of a future state," requires mapping to a meta-model. These projections are a fundamental process for both cognition and characteristics of human memory [Finke 1989]. Based on current literature, in order for one to close the gap between proven psychological models that measure perception and its use in software agents, an engineering method is needed to build the experimental frame to test these projections. Introducing this HBR AP approach as an example allows one to use these projections when comparing the agent behavior of expert perception with that of SOA rule-sets.

1.3. Motivation

Many decisions regarding the development of intelligent software agents are left to the developer of the agent. This results in informal software engineering approaches for knowledge transfer. A method is needed that introduces perception as HBR, using both the accepted practices of psychology and applied engineering. Psychological techniques are able to extract knowledge that is comprised of cognitive elements that are not often found in current approaches used by software-agent programmers and developers. This approach helps to bridge a gap between proven techniques in psychology that measure cognitive elements of task-oriented behavior and applied engineering approaches that seek to transfer knowledge to a reactive agent in a simulated, decision-making environment.

Measures of both human-expert perception and state-of-the-art perception rule sets are readily obtainable. However, engineering approaches for relating these are not as easily accessible.

1.4. Examples

HBR in agent simulations remains a promising example of representing human decision making [Giordano *et al.* 2004]. For successful HBR that combines decision making with perception, verification is required for comparing human and system behavior with its respective simulated behavior [Yilmaz 2006]. This gives rise to a problem domain in regards to creating a logical correlation between human cognition and its simulated counterpart. The domain must also take into account problem-solving skills and perception of anticipated events within a problem set to validate an agent's accurate representation and response relative to the problem task [Butz 2007; Sprague *et al.* 2007]. Thus, observation of expert perception in HBR agents must include a thorough examination of the following theories and concepts:

- HBR
- Agent Theory
- Human Perception
- Visual Cues and *gestalts*
- ACTA
- SAGAT

- SART
- Logic and Agent Formalisms
- Validation, Verification and Accreditation (VV&A)
- State-of-the-art Perception Rule Sets.

Making viable HBR assumptions for a respective agent representation, the AP method incorporates local and partial principles of rationality [Lane *et al.* 2003]. A good relationship between agent models and empirical data implies the verification of the model's accurate representation of the assessment of the process that is being explored. In addition, it should be able to generate a set of observable data [Oaksford and Chater 1994].

Critical decisions based on visual perceptions require event anticipation. It is important to note that, unlike autonomous agent systems, anticipatory systems are not particularly versatile [Rosen 1985]. This creates a problem with handling task-oriented behaviors. These tasks require knowledge of the world obtained from memory, or by reasoning [Pearl 1989, 2000]. However, anticipation consists of intelligent proactive behavior and anticipatory abilities. Without anticipation, a behavioral system can only be reactive in nature [Garrett 2007].

1.4.1 Application Domain for Agent Directed Simulation

Favorable, practical application-domains for using perception in HBR include uncertain domains and environments where spatial constraints are prevalent [Giordano and Brogan 2004, Lowe 1987]. In these environments, expert skills are required to complete critical decision parameters. Extracting expert knowledge as a means for enhancing perception skills would improve HBR decision-making within these partially-observed domains. This knowledge representation can enhance the navigation for unmanned surface and air vehicles as well as serve as a perception interface for robotics [Harvey 2005].

Agent applications presented in this experiment include a simulation of constrained objects where successful, critical decision-making for avoiding objects requires expert HBR performance. Using techniques to extract knowledge from experts provides credible references for defining the agent domain's critical task and all of the unique attributes that make up a critical task. In essence, experts assist with defining the simulation parameters [Chandrasekaran 1999]. This assessment involves collision avoidance in a critical navigation scenario and uses expert navigators to define the agent-directed simulation.

1.4.1 Overview of Methods

Knowledge-transfer from human science to an applied engineering approach is needed to close the gap. Knowledge extracted from experts comprises of the cognitive elements that are not often represented in current, agent-directed solutions. A proof of concept and practical application of this knowledge-transfer by using engineering methods prove feasibility and applicability. The focus and overview of the approach includes:

1. Selecting the experts
2. Validating the experts
3. Extracting knowledge that focuses on perception using ACTA

4. Using game methods to help identify the cognitive elements
5. Establishing SA objects using SAGAT
6. Using gaming to build the critical expert scenario
7. Using gaming to assess a future projected state employing SART techniques
8. Using this knowledge as reactive agents to mimic their behavior
9. Establishing a method to set up the validation case for expert group, control group and state-of-the-art group.

It is possible to conduct experiments that mimic expert behavior based on visual cues by using a method to observe variances between actual human behavior and the associated AP agent. This HBR agent relationship is constrained toward realistic cognitive representation in AB modeling [Giordano *et al.* 2004]. The method introduces perception as HBR within a simulated agent-environment for a particular, human-task correlate to given a "use case" domain [Booch2004]. By eliciting and gathering cognitive data from the "working knowledge" of SMEs for a particular "use case," it is possible to compare the results with the agent behavior exhibiting similar values. Therefore, a correlation seeks a similar task-response behavior by using an example of HBR AP simulation and SME experimental results.

Initially, these combined psychological and engineering approaches provide elements for creating a model that represents and defines input and output variables for a simulation of the task. They also provide test procedures that comfortably allow the use of gaming techniques to develop agents. Simulation interviews compile critical scenario objects used for developing the experimental scenario.

Once cognitive tests are completed, identification of critical objects, and follow-up construction of an agent simulation (symbolizing expert decisions based on expert perception as derived from cognitive tests) are captured, an experiment is performed. The experiment simulates an agent reacting to its constrained environment using values derived from the expert's anticipated interaction among objects within the same constrained environment. This design looks for a statistically significant variance in performance [Law and Kelton 2000]. It is expected that expert decision making represented in agent behavior, based on perception, has a noticeable influence. It is also expected that this influence is observable from the simulation results [Hsiung 1994]. The test procedures are included in the methods for ACTA, SAGAT and SART and are used for extracting data. The Analysis and Results section records the data obtained as well as the results of the tests. Results show that these methods are valid and lend strong support for representing expert perception as HBR.

Results are compared to the human expert, AP, and state-of-the-art rule sets and provide additional relevance supporting the thesis. The analysis focuses on comparisons of the AP agent and validation of the derived the data sets.

1.5. Contribution

An initial contribution is an engineering approach that uses proven cognitive science tests to extract expert knowledge associated with visual perception to improve agents as HBR. These knowledge-transfer methods provide the following contributions through:

1. Classification:

a. Classifying and assigning labels to visual objects identified by experts [Wertheimer 1922] using Cognitive Task Analysis (CTA) techniques [Militello and Hutton 1998]

b. Identifying the physical process associated with perceived objects in a critical incident [Hoffman *et al.* 1998]

c. Recognizing events that result in detection associated with visual cues in a critical incident usable in a software object-oriented environment [Graham 1991].

2. Parameter Estimation:

a. Deriving a parametric description of a critical task conceived by experts to define the physical process and events associated with visual object perception. Using this to build a representative agent task using UML to define known object reference points as identified by experts, e.g., R1 and R2 [Heijden *et.al.* 2004]

b. Using experts to provide an estimate of the parameters and assessment of the uncertainty of the estimate [Tolk 2005]

c. Providing parameter estimation of the agent simulation using psychological methods to extract expert knowledge to confirm that agent parameters relate in a statistical sense.

3. State Estimation:

a. Identifying the conditional state of perception and assigning a UML class label and parametric real value for processes that vary in real time.

b. Providing a Discrete Event System Specification (DEVS) formalism [Zeilger 2000] to ensure that the agent functional structure has definition for:

- Sensing
- Processing
- Outputting.

4. Establishes an expert-derived relation of UML objects in a two-dimensional case and defines simulation reference points associated with the visual objects identified in the AP method.

Knowledge-transfer to an engineering method uses a UML object-oriented model [Booch 2003], game techniques, agent simulation and a generalized V&V approach. Proven psychological techniques that categorize cognitive responses for decisive behavior and applied engineering practices assist with the mapping of these responses as usable software objects [Bonecito 2003]. The addition of gaming provides a tool that experts are able to both use for the parametric description of a critical task and to define the processes, events and visual objects associated with successfully performing the task.

One contribution is that these software objects are reusable. An example is that both the agent simulation and gaming scenario use the same UML objects for referencing critical object data derived from the experts. SMEs verify that both the agent simulation and game scenario accurately reflects their recommended, critical decision-making "use case" [Hoffman *et al.* 1998; Klein 1989]. Using SART, cognitive observations yield a scored SA rating. This application of SART uses the game vignette to represent an expert's chosen critical task as identified in the ACTA. From the experts, SART scores obtained observe an anticipated task response between reactive agents.

This repeatable framework applies a psychological approach and obtains cognitive test results to present a valid use case for an AB simulation. The simulated environment provides critical decision input and output variables for a particular human-task correlate. A central premise is that information gathered through the elicitation of the working knowledge from SME "knowledge sources" for a particular, critical-task use-case should have a similar parallel to an agent's behavior using values derived from SME SART test scores.

The introduction of a method using cognitive test results with an applied engineering approach treats an expert's anticipated object response from a visual scenario, as a "like kind" comparison to baseline human perception [Beer 2003; Goldstein 2002]. It focuses on how close SME perceptions are to the real world, based on their working knowledge of a given scenario, to those in a simulated environment. Using these SME-validated software objects; agent simulations mimic expert solutions and provide a potential for these expert solutions to outperform state-of-the-art rule-based systems.

A contribution is provided of necessary psychology and applied engineering processes to effectively transfer expert choices based on perception to an agent HBR. This thesis demonstrates a method that is repeatable and applicable to other domains as well as within HBR agent domain. Experimental results demonstrate that the method works, as indicated by the comparisons between more accurate expert solutions derived from using the HBR AP example with those solutions derived from state-of-the-art rule sets.

1.6. Dissertation Organization

This dissertation's structure and organization contributes to resolving an applied problem that is associated with capturing the knowledge of human sciences and making this knowledge usable in an accepted engineering approach. An introduction and the definition of the problem domain elucidate the primary thesis statement for the approach. Motivations and methodologies provide a way to test correlates between HBR to human decision making behavior found in natural settings [Bacchus *et al.* 1993].

A literature research discusses principles describing current state-of-the-art HBR agent technologies, psychological models and scientific practices that directly support the capture of human knowledge and its use in an applied engineering method. Second, this paper reviews the gap between contemporary cognitive measures for expert perception with those of state-of-the-art perception rule sets. This includes a review of current cognitive approaches for experimental testing. Engineering-design logic presents an anchor for supporting semantic and syntactic relationships associated with building the correct visual objects for both the experimental scenarios and agents [Braine 1978, Riecke and Heyde 2002]. Supporting information includes a discussion of visual perception, agent domains, gaming and current research specific to the software engineering methods used for agent simulation and scenario development.

The experiment first tests how experts' knowledge enhances perception skills in an agent simulation, and then compares the findings with obstacle-avoidance decisions using AP agent simulation and state-of-the-art perception rule sets.

2. LITERATURE RESEARCH

This literature research concentrates on adaptable methods for transferring knowledge from the human sciences to practical engineering approaches for utilizing expert representations in agents. This review focuses on the domain of agent behavior in relation to expert decision-making and sensory perception for critical tasks. This research also examines psychological models that help identify supporting theories for elements associated with human perception and knowledge extraction. The established procedures for knowledge extraction and related transfer methods found in both the fields of cognitive science and engineering are also observed. Applied logic schemas, model formalisms and state-of-the-art perception examples add to the body of knowledge supporting the method for introducing AP in improving HBR-using agents.

2.1. Knowledge Transfer Methods

A need exists for methods that transfer knowledge from human sciences to engineering for agent simulations human science to engineering [Uhrmacher 2000]. Knowledge transfer implies that there are knowledge extractions that are transferrable to a system. The idea is that reason, and the ability to control it, operate both on a knowledge base and an agreed-upon object-awareness within a common domain [Goldvarg *et al.* 2001]. Examinations of Artificial Intelligence (AI) in agents support the concept that an agent requires "knowledge" about its surroundings, e.g. awareness of other objects and an expectation of their behavior within its environment. An agent may use this knowledge as part of its internal representation of the environment. An engineering approach needs to consider endowing this agent with knowledge by identifying, validating and measuring its reactions as well as its potential for object awareness. This builds upon a shared goal to develop current measures of knowledge that are used in agent-based environments [Weixiong 2000]. To achieve this realism, knowledge measurements and its transferability to an engineering approach must advance coherences between real life scenarios and the simulated agent.

The field of knowledge transfer can be enhanced by merging psychology and engineering. By experimenting and utilizing these two disciplines, researchers may enhance a simulation agent's ability to reason, while also optimizing computational resources used for simulation [Gehlsen and Page 2001]. AP for agents provides a suitable demonstration domain for merging these disciplines, as it allows for a representation of expert knowledge that can be executed as an agent task. Using this domain also provides an observable pathway for extracting knowledge to create an agent by using accepted engineering software methods.

The domain of AP for agents is a good demonstration domain for tasks associated with critical tasks [Klein 1989] because it addresses agent characteristics that can mimic expert behavior associated with conflict avoidance [Holyoak, and Simon 1999]. These characteristics include behavioral functions of communication and actions as related to critical task sense-making (possibly affecting decision making) and perceptions that result in conflict avoidance action(s) within this domain [Moya and Tolk 2007].

Other attributes that make AP for agents an acceptable demonstration domain is that agent representation for perception include "proactiveness and reactiveness," spatial awareness, ability to learn, and social ability. This "intellect" is included in agent behavior and maps to expert behavior [Platon *et al.* 2007]. Further, agents are autonomous and have the ability to carry out many standard programming functions. They may systemically employ perception attributes and the social ability to perform goal-directed knowledge processing over time. This allows agents to function as surrogates to human behavior or as representatives for perceiving other software agents within a constrained environment. For these reasons, agents are the choice for observing expert behavior as HBR.

Complex knowledge is difficult to observe, and extracting perceptual knowledge from experts for agent use proves challenging. However, overcoming these difficulties is accomplished by using effective interviewing procedures, such as those found in Critical Decision Methods (CDMs) [Freeman and Cohen 1998]. This interviewing technique provides an excellent example of how to identify expert knowledge and allow its transfer to a practical application such as HBR. CDM also has a history of effective use [Hoffman *et al.* 1998; Klein 1989; Thordson 1991]. There are many good examples of using agents in this domain, such as applications that are used for observing critical decision-making tasks in natural settings (e.g., firefighting and critical care nursing).

Representing the real world within an agent-based simulation implies that interactions among objects in the model are intrinsically non-linear. In many cases, the concept that these non-linear relationships represent reality in an artificial world is based more a matter of viewpoint than by a physical representation of the reality [Minsky 1975]. Much like the real world, interactive agents have adaptive potential, and their decisions may depend on previous choices made by other agents in the model [Bosse and Jonker 2006]. This supports the idea of injecting valid, expert, cognitive decisions into the model as integral parts of the agent's actions. This allows the introduction of perception into HBR by using valid, expert, critical decisions as inputs for agent interaction with other agents sharing the same state space. Programmable agents mimicking human behavior have explicit simulated responses that accumulate over time. When running simulations faster than real time, these responses may forecast an expert behavior. Agents are not initially capable of understanding the fundamental composition of their surroundings; however they must develop this knowledge through repetitive behavior.

2.2. Psychology Models

There are valid psychological models that provide support for this AP agent demonstration domain. These models provide the capability to identify and extract complex expert knowledge that identifies human perception, observes decision-making processes and provides for knowledge extraction. However, agents as HBR have limited models that use perception to support their behavior [Silverman *et al.* 2006]. Many psychological models simulate the various components of human behavior; however, instances of reactive, agent-validation cases using human behavior remain scarce. Agent theory embraces a very important development in computer science that has recently emerged [Song *et al.* 2007]. That is, agents are autonomous computer-generated programs capable of independent action within both a typical dynamic or unpredictable environment. It is a natural progression to introduce the use of proven psychology models that extract expert knowledge for use within this agent environment.

For example, a typical task would be to map mental processes as HBR. There are examples of success in mapping these processes that include a broad range of applications [Russell and Norvig 2003]. These include areas of scientific research, operational research, manufacturing, network operations, business processes, health care, hospital management, environmental studies, service operations, military, transportation studies, and satellite and space applications [Brutzman *et al.* 2002; Fiswick *et al.*]. These multi-functional uses suggest that it is also possible to merge cognitive tools and engineering methods within a visual simulation to help bridge the gap between proven psychological models and accepted engineering practices.

The psychological approaches for knowledge transfer to HBR include human cognitive testing, task analysis and cognitive mapping to a problem domain. This body of knowledge for cognitive testing provides insight and an opportunity to apply convincing goal-oriented task results from human testing to agent simulations [Chandrasekaran and Josephson 1999]. Understanding these relationships helps associate long-term memory tests that are used within the field of cognitive science to study responses to anticipated behaviors and to utilize methods for validating representations of these behaviors [Wertheimer and Beardslee 1958]. A broad range of tools provide a means to perform a task analysis and cognitive mapping.

One method that combines both task analysis and cognitive mapping of knowledge is Cognitive Task Analysis (CTA). Experimental psychologists help identify knowledge associated with problem solving using CTA [Wertheimer 1922]. This applied *gestalt* theory seeks to identify common objects associated with perception given a problem set. CTA methods are applications used to identify the cognitive skills and mental demands needed to proficiently perform tasks [Militello and Hutton 1998]. Its primary use is to determine the thought processes that users follow to perform tasks at various levels, from novice to expert. However, design engineers use CTA and find it resource-intensive and restrictive. This is overcome by using the ACTA, a variant of CTA, as the preferred method.

Regarding thought processes, a premise for human knowledge assumes that to anticipate events, people rely on mental models and that the human mind has an ability to construct "small-scale models" of reality that it uses [Goldvarg and Johnson-Laird 2001; Johnson-Laird 1993]. This reference identifies perception, imagination, or the comprehensions of temporal events as constructs of a mental model. These constructs may include visual images, but they may also be abstract representations of non-visualized situations [Pylyshyn 2002]. In short, each mental model represents a possibility and is analogous to the structure of the situation represented. These analogies are similar to pictures in the "picture" theory of language and expressed as a representation of causation and reason [Tanenhaus *et al.* 1995]. This mental model is the cognitive layout that a person uses to organize information in his or her memory. It is from this research which studies memory that a realistic transfer of knowledge to an agent may emerge.

There are also valid psychological models that support agents' use of perception. An example with regard to visual perception is a *gestalt* approach that emphasizes the significance of stimulus organizations and relationships [Wertheimer 1922; Wertheimer and Beardsley 1958]. This type of approach does not reject the notion of experience as a precept for human perception. However, the alternative of not using psychological models and using non-experts to develop a problem set unique to an organization of experts would not prove practical for HBR model validation [Balci 2004]. Examining visual *gestalts* obtained from a human expert real world scenario with those created as representation in a simulated environment is more feasible and provides a link for mapping these real-world objects as usable engineering objects.

A *gestalt* approach provides guidance on how visual cues for agent-based objects are perceived [Riecke *et al.* 2005]. This proposes a series of principles regarding visual processing and provides methods to describe (while not attempting to seek a cause) for perception [Hecht-Nielsen and McKenna 2003]. From this, visual objects are grouped into a common environment for test scenarios. This chosen *gestalt* approach supports conclusions that usable logic maps to mental processes. These attributes and their relationships also support mapping of *gestalt* cues [Wertheimer and Beardslee 1958]:

1. *Proximity*: Things that are close together seem part of the same group. Proximity is needed for the development of a simulation and gaming models to trigger-cued responses.
2. *Similarity*: Things that are similar seem part of the same group. Similarity allows for meta-model object grouping.
3. *Common Fate*: This principal extends an anticipated response to a visual representation of the common goal established in a critical task.
4. *Shared Outcomes*: A scenario is able to represent the shared outcome of the critical task.

5. *Appearance of Continuation*: Provides a possibility for applying to both gaming scenarios and agent-based simulations for an appearance of visual continuity.
6. *Perceptual Constancy*: This supports using visual *gestalt*s to experience objects as the same.

This tendency for humans to see familiar objects as having standard shape, size, color, or location regardless of changes in the angle of perspective, distance, or lighting is important. This intuition tends to conform to the object as it is or "is assumed to be," rather than to the actual stimulus [Marr 1976]. Without validation of perceptual constancy, elements of the mental model of a problem-solving task could not assume accurate object representation in agent-based simulation.

Geometries and reactions to *gestalt* objects, as provided by experts, may be used in an AP demonstration domain to address this interaction of both physical and mental spaces. Mental spaces are also the geometries associated with linguistic interpretations of a problem-solving task and continuity remains a main distinguishing feature of physical space [Tanenhaus *et al.* 1995].

As shown, effective psychological methods are able to create visual *gestalts* for perceptual organization [Wertheimer 1922]. This organization provides a motive to search for methods within a problem domain where meta-model parameters can represent *gestalts* for expert decisions-making. Based on these, it is possible to apply critical cues and decision triggers to an AP demonstration domain meta-model and simulate these cues and decision triggers in an agent environment.

Psychological models also support identification and reuse of *gestalt* cues. Regularly, visual cues trigger human behavior [Riecke *et al.* 2005]. These behavioral triggers may be further represented by vectors functioning as "goal triggers" that recognize whether the task of perceiving an object has been completed. These are state transitions [Zeigler 2000]. This lends credence to the use of visual cues that map as *gestalt* cues to trigger a behavioral response based on visual perception within a problem-solving scenario[3] [Yantis 1992].

Regarding visual cues, a pattern recognition schema exhibited in primates, ensures that visual objects processed are truly representative of the visual objects and their responses observed in nature. Observed behavior in primates shows a systematic reduction of the complexity of visual cue stimuli to obtain a minimal pattern recognition cue for memorization [Logothetis 1992]. This mental-mapping process, along with *gestalt* contributions, demonstrates that a primate is generally unaware of its surroundings until object details that are needed to fill in the gaps for the creation of visual cues are completed. Perception and anticipation of a future or past event is initiated by first identifying these meaningful visual symbols.

Agent perception and anticipation are cognitive behaviors also supported by psychological approaches. Some researchers contend that in order for agents to include and represent a concept of perception, recognition of meaningful symbols depends upon

---

[3] Referencing *gestalt* laws of perceptual organization, a *gestalt* cue is refers to size conformity, similarity and surrounding of a visual object.

information used in a stimulus array [Gibson 1966, 1979]. This stimulus array proposes that the environment provides clues necessary for perception. Hence, perception is a direct consequence of the properties of this environment. Thus, visual cues as *gestalts* support a logical mapping to agent-based simulation and expert observations may be mapped as perceptual *gestalt* cues.[4] [Marr 1976, 1980, 1982]. Specifically, the active observer creates additional sources of information and thus problems that are ill-posed and non-linear for a passive observer become well-posed and linear for an active observer. However, an active observer has to deal with much more data, as presented in a stimulus array. Identifying these *gestalts*, cues and triggers allow human observers to participate as active observers and map object perception, as observed phenomena for use in an engineering approach.

Both agent HBR and primate visual systems indicate that one never stops sensing an environment, comparing its content with memory, and constructing appropriate responses. For example, each brain region of a primate's visual system represents a continuous trajectory with a finite number of values in a finite time interval [Logothetis 1992]. Processing stages within different brain areas are highly interconnected and interact in parallel to each other when processing visual attributes. This is very similar to a quantized system allowing for the same behavior as the input and output (I/O) of an internal system [Zeigler 2000]. This homomorphism allows for a preservation of state transitions and maps a state-by-state transition I/O that may be further represented in an experimental visual scenario.

[4] An active observer is an observer engaged in some activity such as motion, focusing, eye-convergence or divergence, head motion, etc., with the purpose of obtaining a better view of the scene.

This homomorphism supports the observation of a visual task response using specified spatial locations with discernable time intervals. Quantization methods reflect a primates' perception of three-dimensional objects elicited through the use of a simplified composition and visual display of these objects.

These definable boundaries support a preface of connected space representation by agents where the mental space would have common boundaries [Giordano *et al.* 2004]. Quantization supports associating symbols within a mental process. This allows for applying mental patterns into a logical process for developing representative simulation objects. These mental patterns are important to identify as they convert to *gestalts* and a simulation domain that reflects an associated cognitive behavior. After observing research that addresses primates' visual perceptions, it is conceivable for perception to have an agent-based correlate as a simulated HBR [Dehaene 1999].

The terms "artificial" and "perception" and, in limited cases, AP, are generally used in the context of computer processes found within the applied engineering and machinery fields (e.g., machine-object recognition and tool-performance applications). AP, used in this sense, does not have an ability to project a future state. This ability is needed to project a future state in order to represent fundamental processes for both cognition and characteristics of human memory [Finke 1989].

Alternatives for establishing perception as HBR include applied systems engineering [Zeigler 2000]. However, systems engineering presents a limited ability for software objects to project a future state. As mentioned, tool performance and machine language generally lack a parallel for cognition. A capability for simulating an object's anticipated behavior is needed for HBR of perception.

For agents, perception in a simulated environment assumes that observers feel, at a minimum, the illusion of being associated with spatial objects and their referential information [Sprague, *et al.* 2007; Stewart 1984]. This includes:

- Shared spatial objects
- Feeling of immersion in a shared environment
- Shared presence
- Interaction with objects or other participants.

When SMEs are interacting within a simulated environment, they anticipate similar actions and reactions to experiences found in natural environments. Table 1 lists these natural translations to perception as identified by Goldstein [Goldstein 2002].

Concept	**General Definition**
Perception	A conscious sensory experience
Perceptual Organization	The process by which small elements become "perceptually" grouped into larger objects
	Series of rules proposed as *gestalt* [Goldstein 1984].
Laws of Perceptual Organization	Specification and logic on how small parts are organized into "wholes"
Perceptual Process	A sequence of steps leading from the environment to the perception of a stimulus, recognition of the stimulus and action with regard to the stimulus

Concept	General Definition
Perceptual Segregation	Perceptual organization in which one object is seen as separate from other objects.

Table 1. Concepts of Perception

These concepts provide insight for a correlation of human visual perception to a surrogate agent-object behavior.

Perception within a spatial model to observe object awareness implies the use of an object's environmental properties as the basis for controlling interactions. Herrero, Antonio, Bedford, and Greenhalgh designed such a model to observe the control of information flow within collaborative agent environments [Herrero *et al.* 2002]. This is an example for increasing modularity of perception and supports identification and development of meta-model elements for the AP domain demonstration. In this spatial approach, governing object interaction for agents employs the following concepts:

1. *Medium* - Similar to simulation Carson's "World View" [Carson 1993].
2. *Aura* - The sub-space that effectively bounds the presence of an object within a given medium, e.g., acts as an enabler of potential interaction. (Agent retains the potential within the model and a similar enabler function is located within the vector trigger(s)).
3. *Awareness* – The main concept involved is in controlling interaction between objects. "Focus" and "nimbus" manipulate the awareness between objects in a given medium. (Part of the set of states (St) that agents of type t can assume).

4. *Focus* - Limiting the "observing" object's interest within each particular medium, e.g., the more an object is within your focus the more "aware" you are of it. (Incorporates a "spatial updating" process).
5. *Nimbus* - Represent the "observed" object's projection in a particular medium, e.g., the more an object is within a nimbus the more aware it is of you. (Provides perception→ deliberation → action representation within agent simulation).
6. *Adapter* – These are artifacts that modify an object's aura, focus, nimbus, and hence, awareness.
7. *Boundaries* – These location parameters affect an object's aura, focus, nimbus, and awareness. (Determine location parameters through pre-set geographic parameters. Agents are confronted either with space constraints, based on these locations, and are forced to respond).

Using agent-based simulation as a cognitive approach to perception infers there will be anticipation of an object's action. This also implies that there is a shared state space with the anticipated object interaction [Chaturvedi, *et al.* 2005]. A supporting observation for agent anticipation is found in the "Primal Sketch" paradigm [Marr 1976]. Within this paradigm, object perception employs a 5-tuple (Type, Position, Orientation, Scale, and Contrast) for the detection of intensity changes, the representation and analysis of local geometric structures and the detection of illumination effects taking place in the process of image generation and representation. Spatial organizations of viewed intensities capture a scene that outlines an object's structure using a set of low-level geometries (e.g. edges, bars, end-points and blobs). A 2 ½ D sketch represents the orientation and depth of the visible surfaces as well as discontinuities. It is composed of some local surface orientation and, as in the previous representation, specifies a viewer-centered coordinate system. In Figure 1, the 2 ½ D sketch represents a conceptual metaphor that shows, in the real world, that not all of the surroundings are seen.

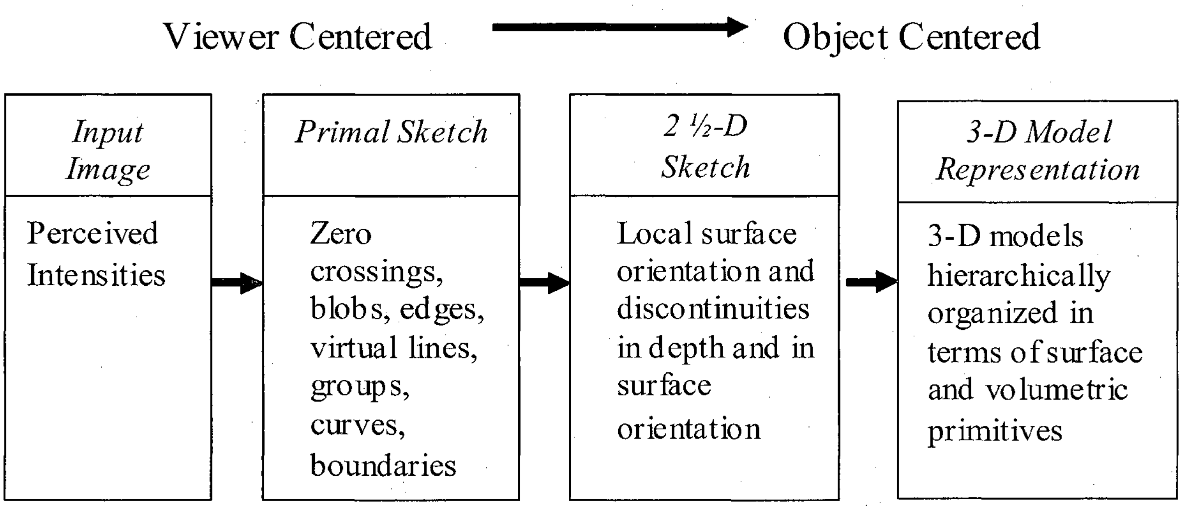


Figure 1. Adaptation of Marr's 2 1/2 D Sketch [5]

Marr's sketch demonstrates a "non-awareness" of all the surroundings where object details become created to fill in the gaps. When these gaps are successfully filled, visual cues may emerge as a precursor to perception. This supports using geometric primitives in the form of UML objects that can represent visual cues for an agent task that projects accurate "real life" representation. These UML objects may be used to construct an engineering approach for experimentation and prove the feasibility of an AP demonstration domain.

[5] An example of this is a person with his or her back turned to you. You may only see half of a body. However, you assume there is some front part to the body.

Both the fields of psychology and engineering provide proven anticipatory models which require observation. This is because anticipated results of a critical problem-set based on human perception is an important attribute in this approach to HBR. Perceptual notions of reality that have an effect on feelings, decisions, and actions are also important attributes in HBR [Silverman *et al.* 2006]. These remain central concepts requiring extensive research and concepts used to resolve intricacies of a stimuli-response-reaction approach to agent behavior not solely based on "spatial" agent reaction.

Anticipatory systems reveal that human perception is dependent on event anticipation [Castelli *et al.* 2000; Goldstein 2002]. In this light, HBR using perception must focus on anticipated events. Current anticipatory systems are essentially "models on models" and are primarily causal (e.g., each agent should have an internal model of its environment, including itself, and by running this model faster than real time it may predict future states) [Zeigler 1986, 2000]. Although current anticipatory systems are alternatives to agent-based approaches, they prove useful references for developing AP architecture. They also offer insight into behaviorally, anticipatory, non-static, goal-directed simulation that includes both elements of perception and anticipation.

Anticipation of objects within a simulation domain suggests that it is a temporal event, where perception of an entity at a time t gives an image of it at that time. At time t, a reference is made to the perception as the current perception (or current image), if there is only one perception. However, at a time t, when it is based on the perspective, there may be different interpretations of an entity, hence, several perceptions. Current images can refer to external perceptions [Sprague *et al.* 2007]. When the current image refers to internal perceptions, then it is about self (or own object) and/or events related with one's own object. These current images may refer to past, current, or future states. In addition, there can be several current images at different times (e.g., $i = 1, 2, 3 \ldots n$;) until future states become current [Croft and Thagard 2002]. This supports a preliminary temporal ontology of AP within a simulation domain including [Ören *et al,* 2007]:

- Perception of an entity at a time t gives an image of it at that time.
- At time t - refers to the perception as the current image - provided there is only one perception.
- At a time t, based on the perspective, there are different interpretations of an entity, hence, several perceptions.
- There are several current images, at different times t_i, $i = 1, 2, 3, \ldots, n$; until the future becomes current.
- Perception is a component of Situation Awareness (SA). SA also builds perception from historical and episodic memory accounts.
- Images of past, current, or future states reflect possibly different interpretations of the current perceptions – they may also create a conflict situation.
- Perception, as a concept, assumes anticipation. AP representations are anticipatory systems whose next state depends on its current state as well as the current image(s) of its future state(s).
- Perception is not limited to "vision" or "retinal" as defined in many robotic architectures but includes observation of both cognitive input and output parameters.

There can also be several images of a certain past based on the points of views of the experts involved [Kirchner 2006]. Current images of past, current, or future states can possibly reflect different interpretations of the current perceptions.

Antagonistic interpretations of the same situation are also considered in mapping mental processes to agents. For example, emotions such as anger may affect the disposition of perception(s) [Silverman *et al.* 2006]. ACTA addresses these interpretations and provides a method to obtain a shared perceived reality for a situation among SMEs.

Agents communicating future actions are anticipatory systems whose next state depends on its current state as well as the current image(s) of its future state(s) or a system determined by a future state [Rosen 1985]. The cause lies in the future. An anticipatory system contains a predictive model of it and/or of its environment that allows it to change state at an instant in accord with the model's predictions pertaining to a later instant [Ören and Yilmaz 2004]. This is a supporting concept for introducing AP to improve HBR. Agent perception must include action that encompasses anticipations.

It is also important to note that anticipatory agents are not particularly versatile. This creates a problem with handling task-oriented behaviors. They require knowledge of the world obtained from memory, or by reasoning [Peterson *et al.* 2000]. However, anticipation is an important characteristic of intelligence and agent proactive behavior requires this anticipatory ability for HBR perception. In order to use anticipation in agents, a forecast of a future state both in an instance of a real task setting and in its simulated environment is needed.

There is an argument that accurate human "sense impressions" are not measureable; however, "sense experiences" that result in a response lend themselves to representations by objects that are calculable [Einstein 1936]. To assume this reality for agents, they must be aware or "perceive" features/objects of their surroundings. Secondly, there must be either an exhibited or non-exhibited response to these stimuli. Some examples of these sense experiences are:

a. *Sensations:* Passive capacity to interact with these forms
b. *Thought:* Active processes of engaging or manipulating these forms and logic to rationalize these relationships
c. *Desire:* Degree of response to the internal state and external environment.

Within the model, these experiences capture a concept of perception that implies a comparison between simultaneously represented objects. Therefore, agent perception assumes the representation of spatial relationships among objects in a coordinate system is independent from the perceiver [Riecke *et al.* 2002]. These spatial relationships have symbols representing visual cues for attributing meaning to an object, so that their processing is actually part of a semantic processing of the associated visual information [Yantis 1992]. For agents, the anticipatory system is an arrangement containing a predictive model of itself and of its environment, which allows it to change state in an instant in accord with the model's prediction to a latter instant [Simon 1990]. In its simplest form, anticipation uses knowledge of future states to decide what action to take in the present. The field of psychology provides us with techniques, such as ACTA, SAGAT and SART, that suspend the state of this sense experience and record from an expert's knowledge of future states, which present actions will affect future events.

An alternative to AI reactive behavior is to embody agents with SA. A generally used application for SA is the cross-check of object geo-coordinates to perceived locations within a same period. The idea is to analyze the differences between actual coordinates and perceived coordinates [Endsley *1995, 1996*]. This generally addresses perception as a holistic grouping of elements and allows for a model to aggregate simulated visual objects as a single cognitive frame of reference. Here, awareness of objects and their anticipated interactions reside within a "volume of time and space, a comprehension of its meaning and the projection of a status in the near future." [Endsley *et al.* 2003]. Although effective at measuring the differences in perceived events, Endsley's approach does not afford a clear-cut path for applying the fundamental foundations of object awareness and perception to HBR perception. However, Situation Awareness Global Assessment Techniques (SAGAT) used in SA studies, which identifies an observer' surroundings, directly supports transferring this expert knowledge to usable software objects. Effective use of SAGAT to create a scenario for assessing SA also supports the use of vignettes to develop both cognitive tests and agent simulation.

2.3. Knowledge Extraction

There are several established procedures for extracting expert knowledge. These procedures are used to ensure a valid extraction of knowledge to HBR and to review the human, memory-prediction function. For authentication of memory-prediction functions, a logical correlation for agent cognition with that of an observation of complex features within a primates' memory is required [Grill-Spector *et al.* 2001; Kamitani 2005].

Fundamental logic that supports memory-prediction demonstrates a relationship to these behaviors within the human framework [Hawkins and Blakeslee 2004].

A premise for both accurate knowledge-transfer and mapping a mental process is that more memory-prediction information gathered from cooperative expert sources yields specific and reliable results. This is important for issues associated with establishing *gestalt* cues used in the method demonstrating AP as HBR. In this instance, the role of "perceptual organization" detects those image groupings that are unlikely to have arisen by accident of viewpoint or position [Sprague, *et al.* 2007]. Once detected, these groupings can map to corresponding structures such as UML objects through a knowledge-based matching process. This is a process of matching expert knowledge to a software object as part of its object-oriented behavior [Hoffman *et al.* 1998]. This makes it possible to use probabilistic reasoning to rank the potential knowledge matches in terms of their reliability. This focus is a search for the most reliable evidence presented in a particular image.

Many human senses retain an ability to resolve fine details. This ability is termed "Sense Acuity" [Watson and McGaffey 2001]. One type of sense acuity is "Visual Acuity" which is a measure of the eye's ability to resolve fine detail. Applying this human concept to an agent behavior allows the agent's ability to detect the presence of another object at a certain threshold without fully recognizing the objects' distinct features. A HBR example of this would be for an agent to detect an object at an increased distance but not allow for absolute recognition of the objects' distinct features.

Exploring these sensory functions leads to a body of knowledge that strongly suggests that, although primate perception is, in part, an illusion, the combination of certain meaningful visual symbols may evoke a cognitive response [Logothetis and Sheinberg 1996]. Using meaningful symbols represented as visual cues supports bridging the human and agent cognition gap. Knowledge transfer takes this body of knowledge and makes it available to engineering and makes possible the correlation of a concept map to support scenario development [Thorsden 1991].

A knowledge extraction alternative to CTA is ACTA, also based on *gestalt* theory. An ACTA gathers information by eliciting the working knowledge from expert sources for a particular critical task. This knowledge maps to a problem domain [Klein 1996]. Fundamental elements of a CTA are incorporated into ACTA. Within this streamlined approach ACTA maintains CTA credibility and allows adaptation for mapping critical task knowledge to agents. Using ACTA provides a means to test similar problem-solving tasks parallel to an agent's behavior.

ACTA provides interview methods to extract information concerning cognitive demands and requisite task skills. These methods allow representation of information via an expert, decision-making, interview structure that translates SA directly to an applied task application. Building on Klein's work, the use of ACTA techniques capitalize on the ease of use, flexibility and clear well-defined results. [6] [Klein 1989, 1996].

[6] ACTA technique developments were part of a two-year project funded by the U.S. Navy Personnel Research and Development Center. The goal of the project was to develop and evaluate techniques that would enable instructional designers and systems designers to elicit critical cognitive elements from Subject Matter Experts (SMEs).

An ACTA critical-task scenario allows for the establishment of experimental problem domains where preferential relationships for SA event choices introduce an opportunity to reveal expert expectation for a perceived result [Likert 1932]. This process inherently incorporates feedback for authenticity of a scenario. These methods provide a template on how to view scenario objects, define a series of principles referencing the visual processing, and provide methods to describe critical events while remaining unbiased and not attempting to seek a cause for the results [Militello and Hutton 1998]. Major parts of ACTA are:

(1) knowledge audit

(2) task diagrams

(3) knowledge audit probes

(4) simulation interviews.

First, knowledge audit assesses expert-novice skill levels by initially engaging experts to recall critical situations where expert decisions were crucial to the outcome of an event. The expert's choice of a critical event and the underlying attributes associated with both expert and novice approaches to the critical task are defined in the knowledge audit.

Second, the task diagram provides an overview of the critical task in graphical format. It helps identify the complex cognitive elements of the task. This supports the notion that objects identified in the critical task may also help define usable symbols for logic comparisons and adaptation to simulation objects for an object-oriented engineering approach [Booch 2003].

Third, the knowledge audit probe contains a series of probes based on expert-novice skill levels in a number of areas, including meta-cognition, mental models, and perceptual skills [Chery and Farrell 1998]. This definition of meta-cognition infers that experts have better self-monitoring skills and are more aware than novices when making errors and better at determining why there might be a failure for novices to fully comprehend the situation [Newel 1990]. Experts appear to have better qualitative "mental models" than novices do. Supporting this notion, experiments found that UNIX programmers, when asked to create a graph that depicted their model or conception of the UNIX operating system, represented higher levels of the UNIX system commands to those of their novice counterparts who only represented lower-level UNIX commands [Endsley 1996]. Another supporting experiment analyzed critical tasks for firefighters' perceptual skills. The findings revealed that experts notice patterns that novices don't [Klein *et al.* 1989].

One alternative to SAGAT is the use of questionnaires. Questionnaires used to measure cognitive responses for SA are generally limited to a specific instance and focus on a problem solution. Within SA, cognitive model referents are inclined to treat perception as a secondary influence of the "world view" [Carson 1993]. Experimental techniques supporting direct perception referents, as found in SAGAT, help identify core elements needed to build HBR perception. A determination of anticipated events and actions of other objects within a domain, as perceived by experts, is needed. This is hard to achieve using questionnaires alone.

Another alternative is to use SART [Taylor 1989]. It provides an assessment of SA of a system or scenario based on an expert's subjective opinion. A scale assesses the degree in which they understand the situation. Advantages of SART include the fact that it easy to administer and it can be used for a wide variety of real-world tasks and simulations. Combining SAGAT and SART may provide an optimum solution tool-set for identifying expert knowledge, assessing this information and their associated objects and providing a realistic framework for knowledge-transfer to software object-oriented engineering approaches.

Another tool used for knowledge extraction is concept mapping. Concept mapping is a technique used for visualizing relationships and organizing relationships between ideas and mental processes. It also provides a graphical tool for organizing knowledge representation. This tool is useful for mapping a mental process for perception that requires interpretive capabilities [Anderson and Lebiere 1998]. This mapping invariably involves a form of sensory object detection derived from interpretation of sensory inputs within the context of a person's "world view" [Carson 1993]. This supports the notion of using established procedures for knowledge extraction and expert identification or interpretation of these sensory objects for use in HBR. Seeking to identify these common objects associated with perception that are given a critical problem set further supports the use of applying *gestalt* theory to engineering logic and approach.

It is also important to map mental processes for perception with *gestalt* cues that can trigger behavioral responses. Insight into this visual cue union with perception is found in the contributions of Marr, who addresses a vision task of deriving shape information from images [Marr 1980, 1982]. Marr's work accounts for how computational systems are able to analyze visual input. In essence, this approach views a vision problem as one of information processing and infers that the agent carrying out an information-processing task has a full understanding capability [Ören *et al.* 2007]. This approach may be categorized into:

1. Computational theory that defines the logic of the execution strategy. This supports conversion of critical visual cues identified by experts into "realistic representations" of a visual cue.
2. How computational theory implements objects in simulations, in particular, the representation for simulation input and output variables and the algorithms used.
3. How representations and algorithms are realized physically. Objects that mimic visual responses realize these physical representations. These support the use of a UML schema to map representative, visual response meta-model to allow for physical representations of scenario cues [Engels *et al.* 2001].

2.4. State-of-the-Art Tools and Methods

There are agent-based tools that make it possible to map human perception used in an applied engineering environment. Use of these tools is essential to build an agent that has "knowledge" about its environment [Weixiong 2000]. Two of the competing engineering trends used for modeling a domain knowledge space are UML (Unified Modeling Language) [Booch 2003]. and DSL (Domain Specific Languages) [Mernik *et. al.* 2005]. DSL is a special-purpose programming language targeting domain-specific modeling. DSL approaches tend to create a new specific-purpose language for every area of interest. Its primary focus is on a particular problem domain, problem representation technique, and a particular solution procedure. UML is a software language able to model possible real-world scenarios as a unified language. It has a proven platform for transforming knowledge to an agent. UML software objects are flexible for scenario development and allow for object reuse. UML, as an engineering approach for knowledge-transfer, uses representative scenario objects in an agent construct to establish valid logical rationale between HBR and agents [Herrero *et al.* 2002]. Agents in this case are native to the approach.

In traditional systems engineering, a knowledge space can be realized by using application-specific syntax (how they talk) and semantics (what they talk about) shared by the agents [Tolk *et al.* 2007]. The Foundation for Intelligent Physical Agents (FIPA) is a central player in the development of these standards for agent communication and proposes a meta-language to describe the structure of knowledge and its content. Communication among agents often changes the environment as much as agent actions do [Searle 1969]. This communication is a foundation for the Knowledge Query and Manipulation Language (KQML) that defines various acceptable communication verbs in the form of performatives. Logical expression that describes the knowledge is part of a Knowledge Interchange Format (KIF). KIF provides a knowledge-content language [Grenesereth and Fikes 1992]. Both KQML and KIF are part of the FIPA-ACL (Agent Communication Language) supporting this meta-language approach. Other examples include the Defense Advanced Research Project Agency (DARPA), Agent Markup Language (DAML), a semantic approach coupled with Ontology Integration Language (OIL) that addresses coordination and composition of web services [Botelho *et al.* 2002].

The use of this semantic and syntactic language structure is found in agent-based tools and methods further supporting the ability to transfer expert knowledge to an engineering application. One example of this is found in a robotic program called "Polyscheme" [Cassimatis *et al.* 2004]. Polyscheme allows for programming of a robotic-action enabling agent to focus on a specific object, e.g., the program function called Requested Foci( ). Upon focusing on an object, the program has the ability to assert "stances" based upon "propositions." This work with "Polyscheme" gives insight about how to address programmatic problems associated with simulating decisions based on anticipation.

With the Cassimatis schema, this "input to" and "output from" specialists that are used respectively, as a time and world argument:

Let o = an object

Let p = a point or location

Let t = a time

Let w = a hypothetical world

Where:

Location (o, p, t, w).

Supporting a method for mapping perception to HBR, this provides for the simulation of immediate and non-immediate states. This work reveals an opportunity for including perception as a part of a similar Foci( ) that allow for casual-based agent interaction. A change in "stance" occurs during critical incident scenarios based on visual cues and incorporates a process identified as spatial updating, e.g., the automatic update of mental representation of the immediate surroundings during the process of self-motions [Hoffman *et al.* 1998; Wertheimer and Beardslee 1958]. This offers a methodology to address updating perceptions in agents.

The Cassimatis Foci() algorithm returns a set of propositions that a specialist (perception and mobility techniques in the Polyscheme model) would like to focus on and enables robotic sensors to focus on a set of variables. This supports an object-awareness logic with a proposition argument: object (o) is located at point (p) in hypothetical world (w) at time (t) is indicated: location (o, p, t, w).

The Polyscheme model uses current state-of-the-art agent actions that introduce perception as an agent for robotic actions that commonly depend on current, past, future, outlying and theoretical situations. Effective robotic sensors must be able to react to an instantaneously-sensed situation of a worldview and a representative state of past, future, remote, invisible or hypothetical states. Polyscheme refers to the latter as non-immediate states and addresses this through expanded time (t) and world (w) arguments. Similarly, the HBR AP demonstration domain addresses these non-immediate states through integrating anticipated scenario results within an agent simulation.

Agent theory also provides methods for dealing with uncertainty associated with mapping human knowledge, as well as multi-phased problems, where the nature of the problem changes as the agent simulation unfolds. This is similar to emerging interdisciplinary work with wicked problems [Basadur 2007] where a goal, based on naturalistic decision-making, is never static and is always changing and Sokolowski's analysis of rapid prime decision-making is observed during simulations and evaluated through varying time sequences [Sokolowski 2003]. An alternative is to provide agent rule-based systems using static goals in an unconstrained environment. This alternative is not preferred since an agent approach with casual and knowledge-based validation operating in a constrained domain better supports the methods and use of perception as HBR.

A good example of an engineering-approach mapping perception and anticipation is found in Multi-Agent Systems (MAS) e.g., multiple interacting intelligent agents. This provides insight into the integration of internal states and positioning of objects needed for agent perception-deliberation-action. Here, a position in the environment defines the situation in which agents are acting. Situation refers to a combination of internal and external events and states [Riecke *et al.* 2005]. This is extremely important for supporting a task representation of the environment.

Another illustration of mapping perception is a model given by Bandini; (Space, F, A). Space S models the environment where: set A of agents is situated, acts autonomously and interacts via the propagation of set F of fields. Space is populated by a "set A" of individuals called agents (a). A is defined by a 3-tuple (t, s, p) where: t = agent type s and agent type t is defined by the 3–tuple; (St, *Perceptiont*, *Actiont*). St = the set of states that agents of type t can assume; *Perceptiont* = $St \rightarrow$ [N×Wf1]. . . [N×Wf|F|] is a function [Bandini *et al.* 2002]. In this example, each agent state a has a vector of pairs in which, for the ith pair, the first element expresses a coefficient to be applied to the field value (*fi*), and the second one expresses the agent sensibility threshold to (*fi*) in the given state. This means that an agent of type t in state s (St) can perceive a field (*fi*) only verified and when the first component of the *i*th pair of the perception function multiplied for the received field value is greater than the second component of the pair. *Actiont* = the set of actions that agents of type *t* can perform:

action: trigger(*s*, *fi*, *s_*)

condit: state(*s*), perceive(*fi*)

effect: state(*s_*)

The field *fi* is active in *p* (*fi*, *Fp*) and agents of type *t* in state *s* can perceive it. The effect of a trigger action is a change in state of the agent according to *s_*. *St* denotes the agent state *p* where *P* is the site of the space where the agent is situated. This model provides a good example of agent domain relationships and interacting perception, action cues and triggers.

Mapping a mental process related to decision-making also suggests a logical mapping of perception [Lane *et al.* 2003]. Research implications are that expert choices obtained by perception fundamentally draw from a comparison between simultaneously represented objects. Therefore, object perception assumes that representation of spatial relationships among objects in a coordinate system is independent from the perceiver. This spatial relationship allows for applying artificial symbols to attribute meaning to an object so that their processing becomes a part of the logical human-behavior connection with the semantic processing of associated visual information [Gaskins *et al.* 2004; Riecke *et al.* 2002]. The ability to apply logic and meaningful symbols for a cognitive process to an agent's interactive behavior, among other agents in a shared environment, helps bridge a gap between observed rational human perception to a stimulus, and agent representation of the same behavior.

Spatial relationships allow logic symbols to represent meaning to an object so that their processing actually becomes part of a semantic processing of the associated visual information. In this instance, *semantic* refers to the application of a meaningful process as expressed and interpreted in a language translation associated with simulated model logic [Ghose and G. Koliadis 2007; Tanenhaus *et al.* 1995]. Using this simulated logic, a model can map objects as a mental process. Research confirms that mapping mental processes involves the creation of a mathematical model of physical space. This is an important referent for AP as a demonstration domain to improve HBR as this logic serves as a template for mental activities mapped onto a mathematical representation of the decision processes within the experiment [Sternberg 2001]. Euclidean geometric models generally provide axiomatic mathematical models of physical space [Newton 1686]. However, when observing the structure of both physical and mental spaces revisiting early ideas on the geometry of mental spaces is in order to provide a more appropriate insight into this unsolved mind-matter dilemma .

In regards to a logic supporting, simulated human behavior, a physical representation involves a continuity of physical space where physical space is infinitely divisible and continuous. For agents to represent mental spaces, it would not be possible to represent this activity as the union of two parts as mental processes inherently do not have common boundaries. This is an important difference in the comparative analysis of mapping physical and mental processes onto mathematical spaces. These are concerning relationships that affect direct mental to physical representation. However, we are able to mitigate this dilemma by introducing a meta-model representation of the associated mental and mathematical processes. This dilemma is very similar to the problems associated with mapping mental processes from psychological models used in a physical space represented by agents. However, timing these processes to a single physical event can be achieved using formalisms.

Formalisms are needed to ensure that an agent's practical structure has definition for sensing and processing and can be used identify output events. An example of this is the DEVS formalism for a deterministic system [Zeilger 2000]:

8-tuple $\langle T, U, Q, \Omega, S, \delta, Y, \lambda \rangle$

T is a "time set" for a continuous system $T = \Re$ (reals) and for discrete time systems $T = \Im$ (integers)

U is the input set; Q is the total state set

Ω is the set of acceptable agent input functions

S *is* a state set

δ is the transition function as defined: δ: S x T x Ω x $\rightarrow$ S

Y = the output set.

λ = the output function: λ: T x S $\rightarrow$ Y.

Hybrid DEVS formalisms are also used to show agent state space dependencies. This includes identifying the advance of time for event triggers used by agents. Figure 2 is a graphical display of a simulation of time advance and event trigger relationships in a hybrid DEVS model with a transitional timeline. The discrete simulation events (t_i)

represent the react, map and execute functions found in the agent simulation used for the AP demonstration domain.

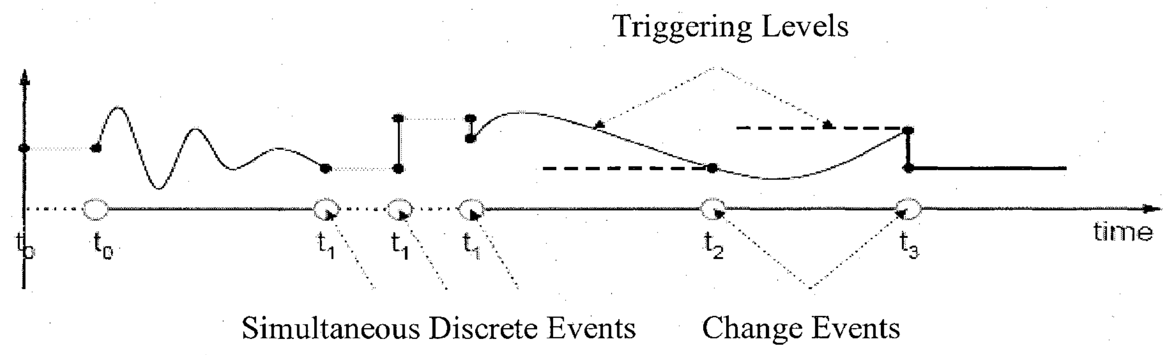


Figure 2. Visual Display of AP Hybrid DEVS Sequence

Another tool used for mapping human knowledge to agents is gaming. Contemporary games provide a current, state-of-the-art agent representation of cognitive behavior. Circumstances in which an agent can only act to capitalize on its utility through anticipating (either deliberately or implicitly within its behavior) the responses to its own action(s) by one or more other agents is defined as a game [Neumann and Morgenstern 1947]. Regarding decision-making, each agent involved in a game is referred to as a player with each player facing a choice among two or more possible strategies. A strategy is a predetermined rule-set that defines what actions are needed as necessary to take when responding to every probable strategy other players might use [Nash 1950]. Experts can provide valuable information for building a decision strategy and identifying its future use as a rule-set for simulation experiments. This also supports the use of games for extracting valuable insight while observing critical decision results along with associated cues.

Also, current gaming software offers an alternative to using pictures or drawings that are traditionally employed by cognitive scientists for identifying *gestalts* associated with perception. Games allow for a visual narrative of events for developing a story line or scenario [Laird 2001]. Gaming objects may also realistically represent natural objects readily identifiable by players.

An example of this is the game Second Life[7] with its immersive, virtual game play and avatars providing human interactions within these virtual worlds. This is a good example of advanced, game cognitive interplay. As with Second Life, an emergent feature of games is the ability for players to build their own environment(s). This ability to build objects and environments as envisioned by the player and the temporal nature of game play allows for custom replication of real-life events. Given a scenario, this replication of events may also serve as a catalyst with players to identify associated important visual objects. Examples of these capabilities are found in the gaming software Quest 3D.® [8]

A review of gaming reveals its adaptability for replicating real-life events and its use as a tool to extract knowledge from experts. Extracting important scenario objects that act as cues for decision-making were favored over the use of visual object graphs or images traditionally used by psychologist. [Thordson 1991]

Alternatives for using AP as a demonstration domain include state-of-the-art agents that operate in uncertain domains or include environmental constraints. These are

---

[7] http://secondlife.com/

[8] http://quest3d.com/

prevalent in autonomous space vehicles, unmanned surface and air vehicles, user interfaces for robotics, and even within industrial and economic process control systems [Maya *et al.* 2004]. These may also represent various levels of a system hierarchy which allow analysis of "natural constraints" or properties of the scenario environments that use a *gestalt* approach [Giordano and Brogan 2004, Lowe 1987].

Current state-of-the-art applications of robotics provide guidance for transferring cognitive responses to an agent. Sensor-object detection techniques used in robotics provide an opportunity to observe rule-sets for sensory responses. In general, robotic entities generally treat perception as an accumulated detection of a sensor(s), object spatial response and as a reaction to an assortment of visual cues. Observations of a robotic approach to object detection reveal an opportunity to include cognitive methods introducing AP. There is motive to align these sensory responses to a more realistic representation of the human senses [Taylor and Kriegman 1998]. Inclusive in this concept of introducing cognitive methods for independent agent action is a prevalent and long-standing motive to create a robotic brain [Harvey *et al.* 2005].

Robots are essentially machines that mimic human behavior and may exhibit autonomous behavior that is capable of independent action within a dynamic or unpredictable environment. Observation of current agent theory, as applied in robotics, provides discovery on how to endow agents with perception knowledge by making use of an object-awareness approach within a partially-observed domain. This adds support for using expert critical decisions in constrained environments to improve representative cognitive HBR realism in agent domains.

Supporting detailed applications of perception are found in recent Unmanned Surface Vehicles' (USV) robotic controls and sensors used for autonomous maneuvering. Each of these demonstrates effective use of applied engineering approaches and state-of-the-art perception techniques. For maritime navigation, these techniques reside within the sensors and software supporting autonomous operation of USVs [Larson *et al.* 2006].

By nature, implementing marine robotics in USVs involves the integration of numerous dissimilar components into a cohesive function. As a comparison for mapping cognitive functions to an AP demonstration domain, multi-component architecture is to the USV what object-orientation is to programming [Yaari 2008]. Like rule-based agents in a constrained and partially observed environment, USVs have competing objectives that balance a final task objective, while at the same time completing near term collision avoidance goals. Effective state-of-the-art USV engineering approaches have the same perceptual challenges as those faced by SMEs participating in this dissertation:

1. *Optimization:* For a USV, collision avoidance must balance task objectives with resource limitations in a constrained environment.
2. *Uncertainty:* There are limits to sensor information and there is no guarantee of actions with oncoming vessels or objects in the constrained environment.
3. *Spatial-temporal relationships*: USV approaches must forecast the location of obstacles as a future event and with anticipation, an assessment made to avoid collision.
4. *Interdependent variables:* In order for magnitude, adding new variables to the USV state space makes decision parameters increasingly complex.

These USV functionalities correlate with the AP demonstration domain because they have similarities with agent choices for critical decisions within a constrained environment. They offer a comparison to agent behavior and provide a reference to proven engineering solutions for a vessel's autonomous control. Also, USV algorithms follow Collision Regulations (COLREGS), providing a marine rule-set applicable to agent decision-making that can be used to compare USV performance with that of an expert agent performance for a similar critical task [U. C. G. Commandant 1972].

Agents are chosen for experimentation over rule-sets, such as those used for perception in robots and USVs, because they are autonomous and have an ability to represent expert decisions and they also have the capacity to directly map cognitive behavior. In addition, agents are preferred because of their proven effective performance using a software object-oriented approach.

2.5. Research Summary

The literature review reveals the gap between current bodies of knowledge for cognition that measure human perception and current engineering approaches for building agents. The field of Psychology has several techniques for explaining and measuring human perception. However, these methods are not integrated effectively into accepted engineering approaches to agent behavior. Similarly, HBR literature reveals few instances of using empirical data to support agent-based simulation that represents human perception.

Research reveals that current state-of-the-art rule-sets that use perception within the field of Engineering do not fully incorporate established cognitive approaches to perception. The literature review confirms that, within the field of Psychology, there are dependable methods for providing functional and accurate indicators of human perception. It should be noted that software developers have not fully used these cognitive models for accurate agent HBR of human perception [Yilmaz 2006]. Further, it shows that ACTA approaches are able to identify, recognize and define representative critical parameters. These can be used to map and successfully mimic expert behavior and event perception in agent-based simulations [Harvey *et al.* 2005]. This agent alternative using an applied engineering approach allows us to capture specific cognitive instances that reference real-life phenomena.

Introducing an engineering approach lends itself toward accurate, simulated decision-making in a partially-observed domain for agent use in robotics, autonomous unmanned surface and air vehicles. Application of these concepts would allow machines to interact with other entities, monitor, react and respond to their surroundings based on cognitive human behavior that has been effectively validated through rigid experimentation.

The literature review demonstrates that ACTA, SAGAT and SART provide vehicles suited for a human knowledge transfer to an engineering approach through discovery, understanding and categorization of SA. An engineering approach makes perception model formalisms for perception a notable contribution for classifying associated human behavior(s) with their representative agent HBR [Goldstein 2002]. Ideally, a method for improving AP and a representative model formalism will lead to increased use of cognitive representation for human decision-making in virtual agents and enhance the scientific and technological methods of agent validation [Herrero *et al.* 2002].

A comprehensive review of current state-of-the-art HBR, agent theory, human perception and established validation techniques support combining ACTA, SAGAT and SART with agent-based simulation development. Circumstances for adapting these methods and establishing metrics for perception require further exploration and discussion.

This research emphasizes the need for a method that bridges the gap between proven psychological models that identify accurate expert knowledge in the form of perception to a more realistic mapping of human behavior to agent simulations. This supports a method for introducing perception-knowledge extraction techniques that provides an agent with expert responses when interacting with other objects in an uncertain constrained domain [Cassimatis *et al.* 2004; Harvey *et al.* 2005; Yeh and Kreigman 1995]. The result of identifying and mapping *gestalt* cues to objects in a simulation and observing the agent interaction and decision-making behavior, could improve both perception and detection algorithms used by state-of the-art autonomous systems. Alan Turing has commented in *Computing Machinery and Intelligence,* "We can only see a short distance ahead, but we can see plenty there that needs to be done." [Turing 1950]. This remains true today with the study of agents as HBR.

3. METHOD AND APPROACH

3.1 Method Introducing Artificial Perception (AP)

Using a method that uses AP as HBR applies knowledge-transfer to an applied engineering approach [Graham 1991]. Proven psychological techniques identify the cognitive objects and mental processes used in this software object-oriented approach [Booch 2003].

The mental process used by SMEs in forming the critical, navigational task result in an expert's "mental model" of the scenario serves as the cognitive test reference [Pylyshyn 2002]. Mental-model processes and symbols affect an individual's overall impression of reality, so these processed symbols represent the context of a cognitive model which allow us to replicate and observe the associated simulated behavior(s) [Carson 1993; Goldvarg and Johnson-Laird 2001]. These processes and symbols are captured and used in a software object-oriented application of the critical task.

As a result of using proven psychological techniques to transfer cognitive symbols, and using objects and processes to a UML approach, one can ensure that methods introducing AP as an example domain for HBR are replicable. Reuse of this method offers a structured approach for validating critical decision making fundamentals associated with HBR.

Individuals differ significantly in their levels of knowledge, such as in the way they represent knowledge as well as the strategies they use when applying their perceptions in a given situation. [Yilmaz 2006]. There remains no well-developed alternative approach to representing this uncertainty in simulation models [Tolk 2005]. Therefore, it is the intent of this experiment to use a cognitive approach to address these uncertainties. A non-obtrusive method is established that introduces AP for representing HBR group task perceptions and expert-situated knowledge in a simulated navigational environment that is independent of individual SME knowledge levels. The tradeoff is between a temporal approach and an inherent expert bias in knowledge elicitation. These are mitigated by separating the knowledge elicitation process into the following segments:

a. ACTA, SAGAT and SART are administered and include critical incident interviews and think-aloud problem-solving methods [Simonin and Gechter 2006].
b. Interactive computer-based gaming scenarios allow SMEs to both interact with the appropriate situation model representation found in the maritime domain and to quickly map the data flow of perceived environments, objects and mental models of the navigational situation [Hecker1996, Laird 2001].
c. A gaming scenario successfully captures the state of Situation Awareness (SA) for the critical navigational task and identifies *gestalts*, cues and objects usable for object-oriented programming [Endsley 2003].
d. Monitoring SME activities allows revision to the scenario as needed to reflect a realistic critical navigational environment for the task.
e. The finial navigational scenario, validated by SMEs, represents a realistic expert mental model that is associated with collision avoidance decision-making.

f. From knowledge elicitation obtained in the previous techniques, a perception meta-model is completed that is characteristic of the maritime domain and strategies that are representative of conflict avoidance for ship navigation [Holyoak, and Simon 1999].

g. Problem-solving strategies are concurrently defined in response to scenario visual cues that are object elements of the mental model for the task. (This ensures that bias does not result from individual predispositions due to scenario prototyping) [Simonin and Gechter 2006].

This procedure provides the first part of a cognitive structure for experimental use. SME critical task "use case" knowledge elicited specifies and categorizes anticipated behavior. This compilation of anticipated behavior is of particular use for comparing agent-object perception with the anticipation of an object's action in a constrained and partially-observed environment.

The following summarizes the initial cognitive approach:

a. ACTA techniques elicit critical tasks from the SMEs [Klein 1996, 1989]. SMEs selected a task of collision avoidance in a restricted waterway. Supporting this task, they chose a scenario of avoiding an oncoming container barge pushed by a tug. This is a situation in which novices would not have the prerequisite skills or knowledge to provide the best decision alternatives to avoid collision.

b. SAGAT identifies SA for the critical navigation task scenarios and the most important visual objects [Ellis 1938].

c. Using gaming software, critical scenario and associated *gestalts* visual cues and trigger objects are built, which are validated by SMEs and mapped to UML as objects.

d. SMES are administered as a completed game scenario. Using SAGAT to assess future projected turns of the container barge and SART for rating a SME's anticipated turn decision of the container barge, the scenario is frozen at a critical decision point and SMEs record a score based on likelihood of each of the container barge turning options. [9]

e. SMEs use a Likert scale (scoring 1-7) with 1 being least likely to occur and 7 representing the most likely to occur. This measure represents anticipated actions of a critical navigation scenario which includes the goal of collision avoidance and an identification of projected future states of the approaching container barge.

f. Bias mitigation training for SMEs questionnaires and scenario development is the final step [Schraagen *et al.* 2000].

Effectively linking critical task methods, concepts and represented objects discovered in the cognitive tests to an applied engineering approach requires effective participation by experts. SMEs from the highly-specialized field of ship navigation on restricted waterways participate in the ACTA. The ACTA identifies hazards associated

[9] Scenario bias is mitigated by adding or removing visual objects (factors) and analyzing results of SME scores during gaming usability tests [Gilovich and Kahneman 2002].

with maritime shipping in this critical environment.[10] Participants provided both identification of the critical "use case" and performed as an experimental population for validating the critical ship-navigation scenarios.

SAGAT supports a method for introducing perception as a means of exploring Situational Awareness (SA). SAGAT using SART scores pinpoint task performance and are insightful factors for effecting scenario-user attention. These scores allow for demonstrating construct validity[11] [Jack 1993; Sloman 1996]. SAGAT methods are reliable and provide an objective measure of SA by directly comparing perceived "human-in-the-loop" simulated events reported as SA to reality.

The method introducing AP builds upon these closely related impressions. That is, observations are validated through the working knowledge of SMEs performing a critical task that requires perception skills. The anticipated responses, based on a mental model, allow for calculating varying degrees of sensory perception [Gibson 1966; Gigerenzer *et al.* 1991; Goldvarg 2001]. SAGAT techniques mitigate timing errors associated with assessing object awareness within an event by freezing a scenario and assessing SME SA of the event. By using gaming techniques to observe this interplay of agent "perceptions" with those comparable to humans, the result is an increased realism in observed object behavior for AB simulations.

---

[10] A professional relationship exists with experienced maritime and merchant navigators. Previous research expertise includes direct experience in the area of ship navigation in restricted waterways and a previous career as a Naval Officer with direct knowledge of the test subject matter.

[11] Construct validity refers to the degree to which inferences can legitimately be made from the operationalizations in a study to the theoretical constructs on which those operationalizations were based.

These SA test events define a singular well-defined environment using verifiable human SMEs. This allows a common perception observation for critical decision deliberation-actions. Based on an anticipated action, a probability of these possible actions in a "situated" environment is calculated and compared [Bandini 2002]. In this case, a situation is a snapshot of the world at an interval of time during which the internal setting does not change and infers that every action statement made regarding the critical navigation task is with respect to a particular situation [Stewart *et al.* 1984].

Supporting this spatial process, a SAGAT technique freezes the critical task game simulation at randomly selected times while observing the game representation of an approaching container barge during the scenario [Taylor 1989]. The time function of the scenario is suspended while experienced navigators, e.g., SMEs, answer questions after viewing the scenario game display [Endsley 1996]. Using SART, the SMEs further score anticipated results about their current perception, comprehension, and anticipated projection of the situation. The questions correspond to their SA requirements as determined from results delineated through a goal-directed task analysis, in this case the ACTA [Militello and Hutton 1998]. These perceptions are compared to a real situation obtained from historical data and expert confirmation. Multiple "pictures" of the navigators' SA are obtained in this fashion and give an indicator of the quality of SA provided by a particular scenario design.

The gathering of SA information using SAGAT and SART techniques provides an objective, impartial assessment of SA that overcomes problems incurred from designing a scenario without SME validation during its development. This SA assumes that group characteristics of stimuli formulate a structure and interpret a visual field or problem in a certain way. This *gestalt* perspective is based on the laws of organization and is afforded an explanation in the context of perception and problem solving [Ellis 1938]. This interpretation of the visual field, and identification of critical event cues and triggers, allow for object mapping to an agent simulation.

Gestalt cues map to agents through the construct of agent triggers from an experimental knowledge base containing two primary types of semantic networks. These networks represent knowledge about the ship-navigation critical-task domain [Ghose and Koliadis 2007]. These are actions and objects that primarily focus on a hierarchy for information flow. A critical, navigational action tree demonstrates this object class hierarchy (Appendix, Figure A-4). The tree relations specify observable traits of each particular phase used for assembling causal reasoning for the HBR AP approach.

The basic plan structure of the action hierarchy is the action pattern, which executes a set sequence of behavior [Firby 1987]. These action patterns significantly reduced the possible combinations of goal-oriented action selections for object perception within the tested critical-incident scenario. Each choice yields a result that could be represented as an object within an object-oriented model [Graham 1991]. As shown in (Appendix, Figure A-4), these perceived objects, M-1 through M-10, were representative of the UML objects used to construct the agent-based meta-model architecture using AnyLogic ™. These perceived objects represent objects and visual cues that may be mapped to UML.

This action hierarchy shaped a prioritized collection of scenario elements, the behavior of which converged on the particular goal of safe navigation to avoid the oncoming container barge. This represents a multi-tier reactive plan that consists of a set of elements, which may be either perceived action patterns or scenario objects identified by the SMEs in the ACTA. Not only are they prioritized, these elements also have *gestalts* that are represented by vectors and that function as "goal triggers" which may include specific cues, recognizing whether the task of perceiving an object has been completed.

The visual cues identified in the ACTA and depicted in the action hierarchy primarily assure that the elements used to create the UML representation of the critical scenario *gestalts* in both the gaming scenario and the AB application were indicative of the actual cues found in a "real life" natural event. The initial perception cues and their triggers expand the AB application to include a decision matrix for anticipated vessel movement.

Applied logic and meaningful symbols represent this cognitive process for an agent's interactive behavior among agents in a shared environment. These present a means for observing rational human perception to a stimulus as agent representations of the same behavior. This suggests that cognitive tools and behavioral tests merge using simulation to help analyze the gap between cognitive test scores and AB behaviors. These methods link human and agent cognition, thus providing a validation case for HBR AP in a simulated environment.

The experimental method encompasses diverse agents operating in a complex system that are able to evolve through time and space [Riecke and Heyde 2002]. Empirical data resulting from the methods introduced for improving AP can be interpreted by the emerging behavior resulting from agents' recurring interactions within a constrained environment, thus representing the critical-decision aggregate state space.

3.2 Proof of Concept/Feasibility

The HBR approach for improving AP draws on proven psychological models and cognitive experimental methods successfully employed in both the U.S. military and civilian sectors [Klein 1996]. These methods provide established means for accurately accounting the human behavior associated with a task experience [Militello and Hutton 1998]. To accomplish the experimental objective, consolidated SME interviews and survey information are synthesized into cognitive demands that contain the fundamental elements needed to construct a critical task [Annett 2000]. This provides a means for obtaining and identifying the following tasks that are essential in creating the critical navigational, task-cognitive demand elements:

- Identifying the difficult cognitive elements
- Explaining why the task is difficult,
- Identifying and listing experienced SME Cues
- Identifying and listing the experienced SME strategies.

Based on this review, applicable psychological instruments and an adaptation of Wertheimer and Klein's work using critical decision methods call for collecting cognitive data in a natural setting [Klein 1996; Wertheimer 1922]. The following are five decision characteristics as specified:

1. Provide an interview format for eliciting cognitive elements of the navigation

tasks;

2. Provide a representation format that presents data in a symbolic logic;
3. Highlight the aspects of the critical navigation task that requires expertise;
4. Reduce the time it takes to elicit the critical task and develop task diagram;
5. Provide a survey format with a relative ease of use.

Critical-scenario events identified by SMEs and validated by means of their expert knowledge support a hypothesis solution. Psychological models used include ACTA and cognitive probe techniques for extracting these events, identifying the associated visual cues and allowing proficient V&V of intended and specific uses for the HBR AP model. Results are manifested into an identified goal-directed behavior [Chandrasekaran and Josephson 1999]. In this case, the goal is a behavioral task of safe navigation in a restricted waterway when confronted with uncertain anticipated movements of an oncoming vessel.

The initial step in the process is to select and validate the experts. A knowledge audit worksheet records the experience levels of the experts (Appendix, Figure A-2). Inquiries used to help validate this experience also include the question of whether they consider themselves to be proficient ship handlers. In addition, the participants list their years and months experience in ship handling, as well as the amount of time they have spent in navigating ships under critical situations in restricted harbors or waterways. This information becomes significantly important, since experts have different backgrounds and experiences that may influence their reactions and result in various responses when faced with a critical-avoidance task. Validation of experts as decision-makers in an expert system then becomes a question of capturing a true sampling that represents the broad cross section of the profession.

For establishing a task reference, 6 of the 22 total navigators participating in the experiment, with each having over 25 years of experience, initially respond to basic questions of what may constitute a critical-navigation task. In this instance, the experts chose responsibilities associated with safely navigating a restricted channel, or harbor, as these chosen references are cognitively complex. These tasks are those that SMEs identified as difficult to perform, which are often done incorrectly and require expert training [Deutsch and Pew 2002].

Next, using ACTA, interview methods extract information concerning cognitive demands and requisite task skills. These methods allow representation of information via a structure that translates more directly to the applied, critical, navigational task application. In this case, it is the use of the experimental scenario and SME interface development recommendations. Building on Klein's' work, the use of ACTA techniques capitalized on the ease of use, flexibility and clear, well-defined results[12] [Klein 89, 96].

First, elicitation and selection of a critical group task is completed. This elicitation identifies a specific environment with a specific stated goal for the task [Chandrasekaran and Josephson 1999]. As provided in ACTA procedures, SMEs provide a list of perception skills expected for the critical action scenarios using Knowledge Audit

[12] ACTA technique developments were part of a two-year project funded by the U.S. Navy Personnel Research and Development Center. The goal of the project was to develop and evaluate techniques that would enable instructional designers and systems designers to elicit critical cognitive elements from Subject Matter Experts (SMEs).

Worksheets and Probes (Appendix, Figure A-2, and Table A-3). Using probes, SMEs achieve the ACTA objective of revealing situations, and as well as objects, observed responses that are used as reference measures for validating a computer representation and resultant behavior for the AB simulation.

Second, knowledge audit interviews refine the level of expertise and provide fidelity for the cues and strategies associated with observing a tug pushing a container barge in a restricted waterway (Appendix, Table A-3). During the interview, SMEs define aspects related to the expertise required to successfully complete the navigation task and also define the aspects which make the task difficult.

Following the knowledge audit interview, a simulation interview identifies a task's cognitive elements within the context of a specific incident that SMEs identified as hazardous navigations in a restricted waterway (Appendix, Figure A-3 and Table A-2). This interview accesses an experienced navigator's expert-thought processes during an important and critical scenario. It also helps to identify important events, decisions, and judgments and an analysis of these events. The simulation interview also facilitates identification of possible novice errors [Deutsch and Pew 2002]. Using the critical-task diagram, a simulation of the event represents the ship navigation scenario. Quest 3D® simulation game software [Quest 2006] is used to replicate the scenario (Appendix, Figure A-6).

Simulation interview worksheets also provide knowledge audit input for both creating the SART test using "Ship Simulator" gaming software and UML for the AnyLogic™ AB simulation [Camerer 2003]. The resulting goal chosen by the SMEs is a task which results in safely navigating a restricted waterway, based on the anticipation of the approaching vessels' movements (Appendix, Figure A-3).

SMEs decompose the task into subtasks by thinking about how to react when confronted with the scenario. A whiteboard helps to facilitate the identification of the navigational task's elements and to record the findings. Once identified, the motive is to reduce the tasks into less than six, but more than three, steps [Thorsden 1991]. Each subtask is listed from right to left across the whiteboard using arrows to indicate the order in which the steps occur.

Subsequently, SMEs determine which subtasks require cognitive skills. The explanation of "cognitive skill" is specified to the participants as skills that require judgments, assessments, problem-solving and thinking ability [Schraagen *et al.* 2000]. SMEs identify and circle those tasks which require the most expertise on the whiteboard as shown in Figure 3.

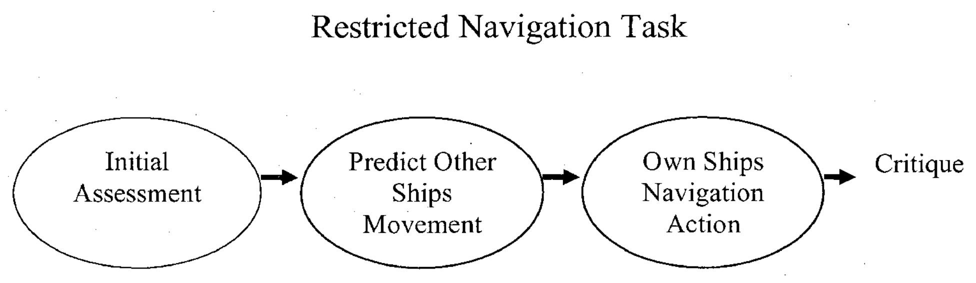


Figure 3. Navigation Task Flow Model

An overall view of the critical navigation task emerges and an accurate assessment establishes the areas where the complex skills are needed to accomplish the critical task. A focus remains on the most important subtasks that are identified as requiring cognitive skills. The result is the creation of a critical task diagram to assist with establishing a focal point for completing the remaining steps of the ACTA, and which is accomplished in the same manner as the higher-level task flow diagram depicted in Figure 3 above. The critical task diagram in Figure 4 depicts how to conduct the knowledge audit.

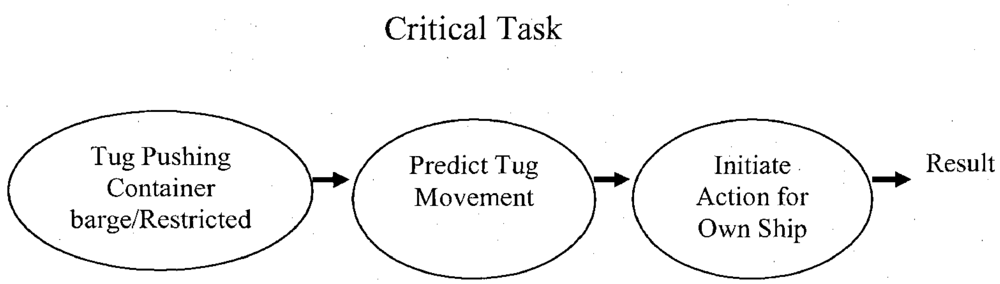


Figure 4. Critical Task Diagram Flow Model

The ACTA knowledge audit becomes a referent for surveying the key aspects of navigational expertise. It contrasts expert-novice knowledge on the following areas:

- Perceptual skills
- Past and future
- Big picture (explains) or mental model [Johnson-Laird 1993]
- Job smarts
- Improvising
- Self-monitoring.

Predicting tug movements and initiating ship navigations require difficult cognitive skills [Riecke and Heyde 2002]. These cognitive skills are listed at the top of the whiteboard. Underneath the task, three columns outline the following headings: (1) Example, (2) Cues and Strategies, (3) Why Difficult. Starting with perceptual skills, each of the three columns on the whiteboard completes the appropriate probe for each of the knowledge audit categories as shown in Figure 5.

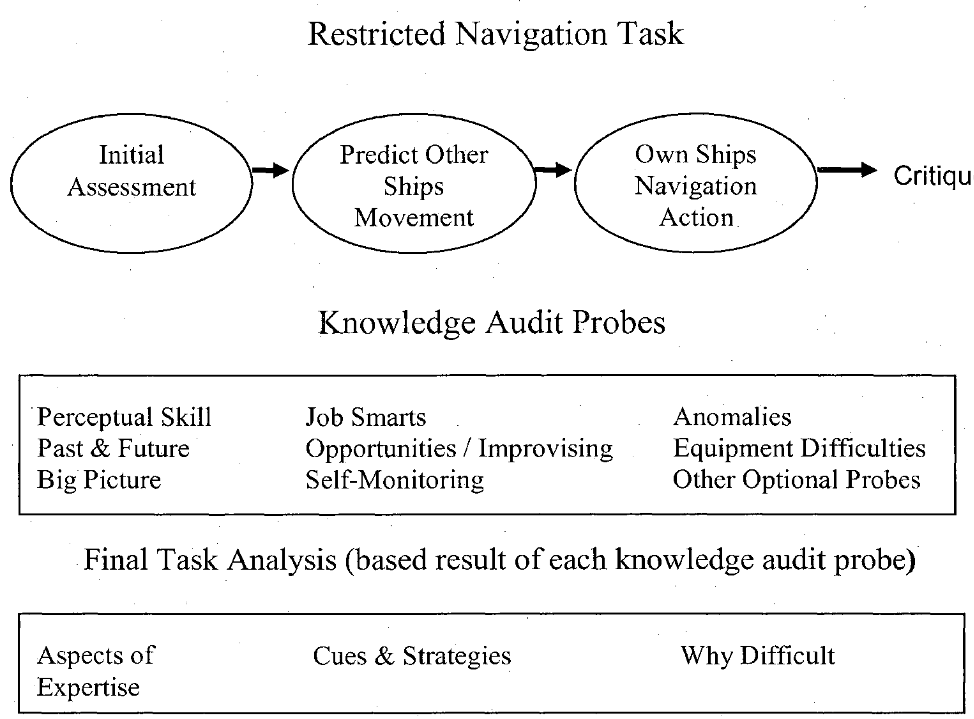


Figure 5. ACTA of Restricted Navigation Tasks

Initially, the participants recall a critical navigational situation that required effective ship handling and ensured safe harbor passage. These navigators recall any "unusual" decision-making that may have been required to complete this task.[13] A conceptual graph analysis outlines this process (Appendix, Figure A-9).

The ongoing assessment of perception skills uses an approach to align a SME knowledge base to the critical navigational tasks identified using ACTA. Associated risks deal with comparing abstract behavioral representations and associated natural events [Bonecito and Burns 2003]. Successful identification of accurate representations is through the observation of the "closeness" of responses between an actual event(s) and a simulated event(s). Specific challenges related to associating human behavior with a surrogate representation of this activity included [Chery and Farrell 1998]:

- Very high inherent complexity
- Numerous nonlinear relationships that have the potential to interact chaotically over many different orders of magnitude
- Complex coupling with differing parts of a simulation system.

Using a navigation expert in group (aware-think-response) logic, the procedure includes selecting a group theme (use case) within a specific environment, surveying for information peculiar to the use case and experimenting with related ship navigation scenarios and vignettes[14] [Braine 1978; Grassi 2000]. This validates that the cognitive model is responding similarly to a "real life" perceived event [Stewart *et al.* 1984]. Key procedures for concept development include:

[13] SMEs were consistent in the critical cues they recalled [McNeese and Vidulich 2002].

[14] The procedure included asking a trained ship navigator questions regarding his or her perception of an anticipated vessel movement during restricted harbor or port navigation, e.g., what do you think the vessel is going to do in this particular scenario?

1. Design a theoretical framework to observe critical navigation task mental models [Gigerenzer *et al.* 1991].

2. Elicit a mental model representation via SMEs for an AP model and simulation development. Representation validation use a series of interviews, surveys, and experimentation and simulation ANOVA tests with ship navigation experts referenced as experienced in the critical task.[15] Each process validates and checks for consistency [Lowe 1987].

3. Utilize components of the scenario context, SME definition of a mission critical task, and individual expert testimony that forms predictions, or a recommended type of mental model to accurately recognize approaching vessels movements, thus directly contributing to model validation [Simon 1990].

4. Develop method validation and test measures [Jack 1993].

Regarding perceptual skills, SMEs detect patterns and make discriminations that novices cannot see. These eventually become the building blocks for *gestalts*. The results are recorded under "Aspects and Perceptual Skills" for each strategy in the knowledge audit interview (Appendix, Table A-3). Next the SMEs are asked, "Why is this task hard for novices, or why don't novices know to do that?" These responses are then documented under the heading "Why Difficult." After completing the information for perceptual skills, probing continues for each area of expertise using the same approach. (e.g.,

[15] The goal was to define a set of "scenes" that could be combined and elaborated into a mental model representation for simulating a real world event. Another objective was to develop rule sets and probability tables based on validation of these model representations.

obtaining the example, canvassing the SMEs on why this was difficult, and then soliciting the related cues and strategies). The critical tasks are extracted from the task diagram and annotated as appropriate across the top of the whiteboard. The perceptual skills are also probed.

A concept graph is created that visually illustrates the internal knowledge structure of the ACTA based on the primary goal of safe navigation in a restricted waterway (Appendix, Figure A-9). These goal actions consist of nodes interconnected by their common action vectors. The nodes represent action items, goals, concepts or events that are needed to complete the primary goal and provide a reference for validating the HBR AB scenario development.[16] In this hierarchical structure, nodes contain either single concepts or single statements. This method is similar to concept mapping, but it includes a formal and detailed collection of nodes, relations, and questions [Pace 2001].

There are specific relations for each type of node and a set of formal knowledge probes developed for each node type. The critical navigation hierarchy results are recorded on the concept graph. This identification of important navigation cues becomes important for mapping navigator cognition to UML objects when building an agent representation for the simulation. Next, each expert continues the analysis using these probes to define a deeper level of information for the graph. More so, they include a third stage of validation for the conceptual graph by having a group member rehearse task perfromance to check for omitted information [Jonassen *et al.* 1999]. These help identify important visual cues that are mapped to UML objects.

[16] The primary goal is the first step of the concept graph and is labeled as #1.

The utility of SME cognitive data as a referent in the knowledge audit helps support a valid construction of introducing AP into HBR. Patterns are discovered while constructing cognitive graphs that help define both the critical *gestalts* and the cues that trigger critical decisions. Regarding agent composition, the question arises, "What suffices in HBR as an accurate representation of the cognitive behavior via an agent and how may this be confirmed?" First, there is an enduring argument that there are no occurrences of an absolute valid model [Sterman 2000]. However, there are generally accepted practices that the credibility of agent models enjoins with their "intended use" and that accepted simulation is a prescribed condition under which the model is tested. In this case, the practices are prescribed and identified during the ACTA.

The use of game building facilitates SME creation of a realistic scenario. SME interaction helps build game scenarios that represent the critical navigational task. SME engagement in building game elements visually confirms that the navigational visual cues identified in the simulation interviews are realistic. Game scenarios represent the critical navigational event identified by the SMEs in the ACTA. This involves gaming interviews with experienced ship navigators who identify scenario cues and *gestalts* that become a basis for expert system and state-of-the-art comparison. As shown in Figure 6, experts provide task input and output parameters in order to accurately replicate *gestalts*, triggers and decisions associated with task behavior.

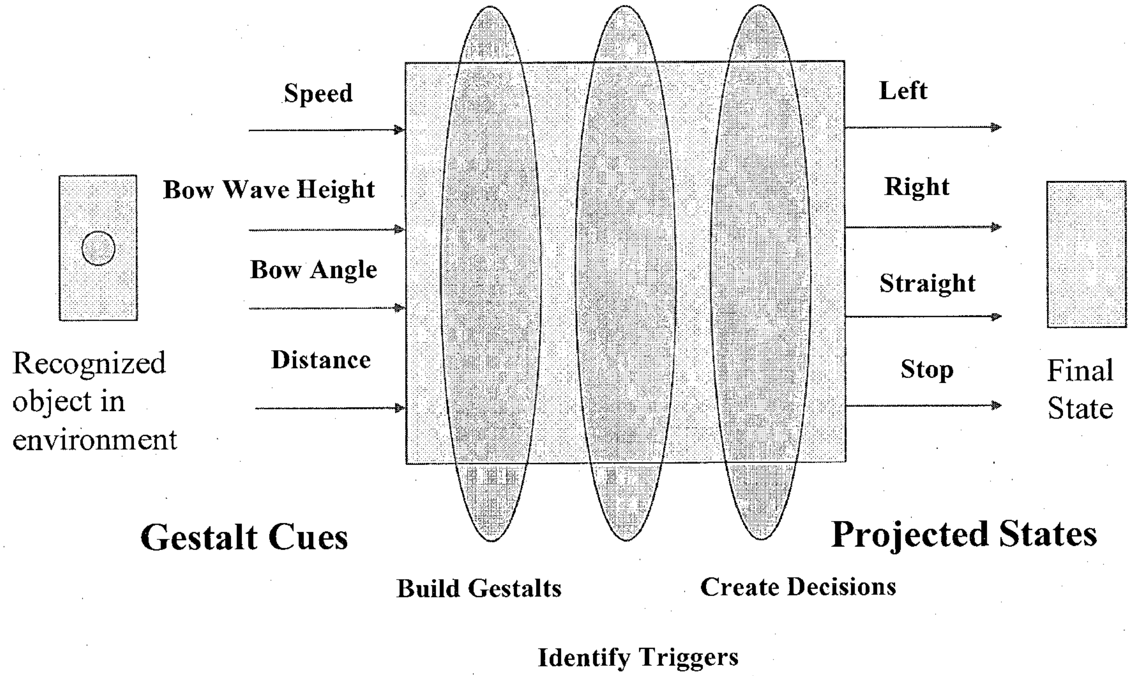


Figure 6. Example of *Gestalt*, Trigger and Decision Input and Output Parameters Developed by Experts Using Gaming.

Game objects elicited from the SMEs assist with building realistic scenes and scenarios. Critical Incident Techniques (CIT) help experts build these scenarios [Freeman and Cohen 1998]. The SMEs recall a historical account and discuss a specific navigational incident that is of particular importance within the same context of a restricted navigational task. The goal is to obtain a complete account of an expert's solution toward solving this task as well as to identify factors that may influence the design of the agent simulation. One risk associated with using this method is that much of the knowledge elicited may be idiosyncratic or atypical [Annett 2000]. The following knowledge elicitation approach helps mitigate these risks.

When building the storyboard for the gaming vignettes, SMEs provide feedback about possible navigational situations and scenes that accurately depict the critical task. This helps facilitate scenario development and adds to the "realism" of subsequent AB simulation. In addition, when building the storyboard, experts refer to the cognitive graph and probe to further identify *gestalts*, which play a critical role in recognizing existing hazardous environments. In this scenario, they included the range and bearing from the approaching container barge, wave height of the barge bow and speed estimates. Associated cues are recorded from these *gestalts*. Parameters and thresholds which could trigger each cue are then established.

The game-developed navigational scene, as selected and verified by SMEs, is principally the recommended mental model of the critical task [Finke 1989]. The mental -model choice represents the criticality of the anticipated behavior and provides projection of a future mental state based on the participant's current perception(s) [Iarossi 2003]. Via SAGAT measurement methods, the participant's perceptions are accurately tested and important information related to the critical task is gathered.

Games are also used for simulation interviews. Viewing the game, the first groups of SMEs[17] verify that the simulation is an accurate depiction of the critical task. Participants review the gaming vignette and imagine they are performing the navigational task at hand answering the research inquiries about navigational decisions and judgments used for the scenario [Chandrasekaran and Josephson 1999]. The whiteboard has five rows with the following headings: (1) Events, (2) Actions, (3) Situation Assessment, (4)

[17] This initial group of SMEs consisted of 6 ship navigators of the 22 participants, each having over 20 year's ship-handling experience.

Critical Cues, and (5) Potential Errors. After SMEs review the vignette, they reminisce over the situation and list the major events, judgments, and decision points that occurred during the incident.

The major events are identified in the first row; then, the four remaining rows are completed as listed below in Table 2. This process allows for collaboration or identifies needed corrections and provides further verification for the experimental AB simulation development [Militello and Hutton 1998].

Events	Review the scenario and list the major events that include judgments or decision points. List these on the white board.
Actions	As the navigator in this scenario, what actions would you take to safely avoid the tug and container ship?
Situation Assessment	What is your assessment of the scenario?
Critical Cues	What visual cues allowed you to assess this scenario?
Potential Errors	What are the mistakes that an inexperienced navigator would make in this scenario?

Table 2. Method for Extracting Simulation Interview Probes

Figure 26 lists sample results from the simulation probe.

Scenario cues and objects map to UML objects directly representing the critical scenario components. Using these UML objects to configure agents and scenario objects for use in AnyLogic™ agent simulation allows for a logical connection between SMEs, ACTA results and agent simulation. Spatial relationships and perception formalisms from MAS are referents for mapping critical cues and scenario objects to UML. (Appendix, Figure A-3) Validated by SMEs in AnyLogic™ as a realistic representation of the gaming scenario, terrain features given to spatial restrictions accurately represent a restricted waterway. At simulation run time, agent-objects interact within this constrained environment and require actions to avoid collision as they compete within this restricted geospatial environment [Metaxas and Terzopoulos 1993; Meyer and Szirbik 2007; Platon *et al.* 2007]. Pre-programmed agents implement each critical decision model object from the ACTA and the agents react according to their view of the world [Hogeweg 1999].

Using the gaming scenario identified in the ACTA, experts map model objects against their knowledge base for the critical navigational tasks SMEs and validate the "realism" of the ship objects, the restricted channel environment, and visual cues identified in the simulation interview.[18] This includes defining what constitutes the uncertainties of an oncoming vessel's movement (Appendix, Table A-4). This forms the basis for conducting the cognitive experiment using SAGAT.

The method to introduce AP for improving HBR relies heavily on SA as an approach to interweave spatial cognition and object recognition with reality checks and balances. SAGAT obtains navigational expert SA for evaluating SME observations and gathers test results [Endsley 1996]. These techniques assess the anticipated result of a future event within a critical navigational situation through a periodic and random

---

[18] A *gestalt* approach identified visual cues by SMEs.

freezing of the gaming vignette. A series of questions are provided to the game viewer to appraise knowledge of what event may happen in the future when faced with the difficult navigational circumstances at the time of the "freeze." Upon pause of the vignette, a numeric score captures this knowledge assigned by the SME based on their probability estimate of the anticipated approaching and endangering container barge movements.[19] SMEs use Situational Awareness Rating Techniques (SART) to score this measurement of the future anticipated event when the vignette stops.

These measurements provide valid indicators for what is truly happening in the scenario at the time of the freeze [Stewart *et al.* 1984]. This procedure also permits unbiased data collection and establishes a baseline for HBR perception measurement. There is also the advantage of applying this measure to the AB "UML created" perception objects at run time in the test simulation [Hill *et al.* 2002]. SAGAT directly supports utilization of SART to score this measurement of the future anticipated event when the vignette stops [Taylor 1989]. Experimental SART tables record the results (Appendix, Table A-12).

SART scores provide probability markers for assessing SA based on an SME's subjective opinion [Kamitani 2005]. Anticipating an impending container barge action, participants rate on a Likert scale (1through 7) from least likely (1) to most likely (7) the probability of following anticipated events:

- Vessel will continue on present course
- Vessel will alter course right (starboard)

---

[19] Possible approaching container barge movement options previously selected during the ACTA.

- Vessel will alter course left (port)
- Vessel will stop.

Listing these probable factors, SMEs complete SA questionnaires at the conclusion of each scenario. These are used to rate their awareness of an evolving situation on a seven point scale [Weixiong 2000].

One benefit of creating task specific scenarios using these techniques is the ability to incorporate a wide range of task types in simulation validation tests for most "real world" critical scenarios [Tanaka 2005]. Using Likert scales, a summation of item scores allows for analysis of numbered responses to alternatives for individual events.[20] To distinguish individual events from the summated Likert scale, "Likert-type" items reference the individual items. This technique presumes the existence of an underlying latent or natural variable whose value also characterizes the SMEs attitudes and opinions. If it were able to measure the latent variable directly, the measurement scale would be, at best, an interval scale.

A score obtains the likelihood of an anticipated response expected when confronted with a critical task [Likert 1932]. SMEs review the gaming scenario and rank the most probable event they anticipate will occur when the scenario stops. The events are predetermined by the SMEs as safe alternatives for collision avoidance. SART methods provide the test scores [Taylor 1989].

---

[20] Likert is explicit that a numerical scaled response is an alternative to the actual scale. Although these responses are representative of a scalar value when coded, they are not a summated Likert scale [Likert 1932].

An agent model is needed to introduce AP as HBR. For building an accurate representation of a critical task, a capability is also needed for agents to perceive the environment and the ability to map this perception to an appropriate meta-model [Zeigler 2000]. As shown in Figure 7, *gestalt* cues help build the critical task meta-model and identify projected states that represent critical decisions.

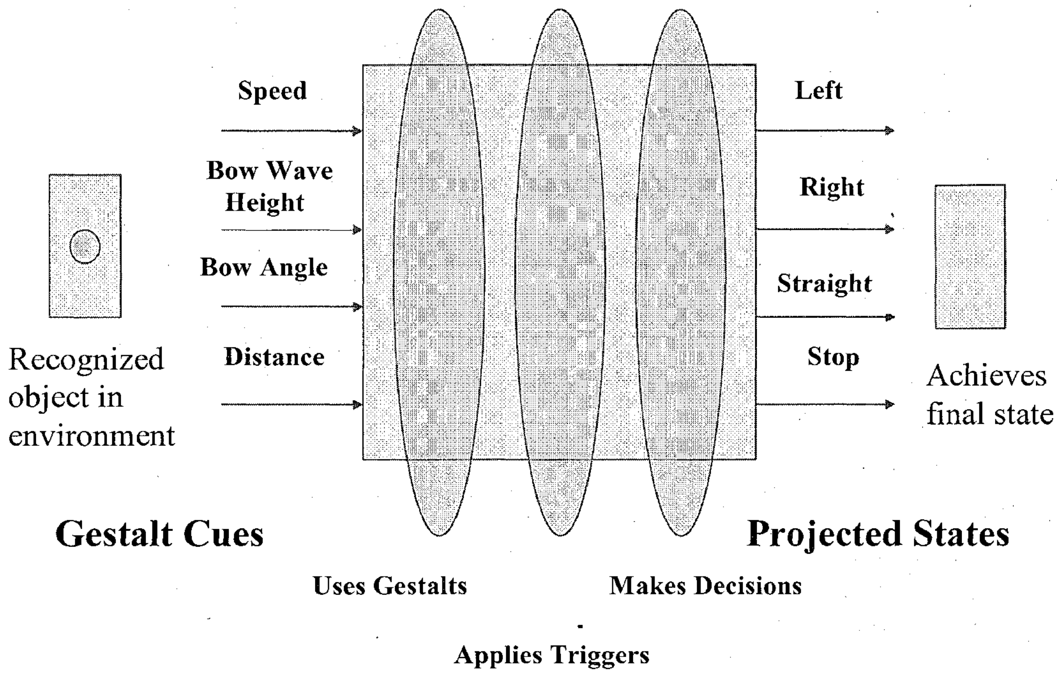


Figure 7. Example of Using *Gestalt* Cues to Build a Meta-Model for Projecting Future States

In conjunction with the ACTA and game development, the conceptual, mathematical and logical representations of scenarios are created which mimic the real scenario (Appendix, Figures A-8 and A-9). An agent perception simulation is concurrently developed using agent logic software [AnyLogic™ 2006]. For the ACTA, gaming scenario and simulation, SMEs validate cues, model object constructs and logical layout of the critical scenario. Then, specifically identified *gestalt*s map as UML objects, forming a logical connection between SME ACTA results and the simulation objects.

Experimental scenarios are developed in Quest 3D® VR software and then integrated into Ship Simulator gaming software [Hecker 1996]. These game simulations further verify realism by using SME results from the simulation interviews and identifying g*estalt*s (Appendix, Table A-4). This approach observes that HBR perception agents respond similarly to real-life events [Sprague *et al.* 2007]. The general video file formats developed by means of QUEST 3D VR software used in the experiment incorporate the navigational scenarios for the test simulations [Quest 2006]. AB simulated SME HBR is predicated on critical, navigational, experimental data [Joines *et al.* 2000].

Model logic is applied for an agent addressing projected states that maps to a metal-model used to build the agent simulation. Anticipation logic for agent-object behavior given is as follows:

Let R_0 represent a part of reality that the agent interacts with;

Let R_i represent an internal description of the agent's environment R_0 and of the agent itself;

Let M represent an internal interpretation (i.e., a model) of this description.

Therefore:

M allows for the interpretation of a number of SME realities, R_i's are made by the agent-object in order to anticipate a future event. This allows M to express a number of comprehended realities, R_i's. Causal variations are due to the different possible events in the environment, and are influenced by interactions between the environment and the agent itself. Thus, an agent may "anticipate" a number of different R_i's, each giving different predictions of the state of R_0 at some future time. An important point to remember is that the agent maintains the potential directly affecting an environment. This also must take into account the anticipation of a future event, as follows [Ekdahl 1994]:

Let e represent a vector of environmental states

Let s represent a vector of past states

Let G represent a function that generates the future states.

Therefore:

$$\langle e, s \rangle \Leftrightarrow \overset{G}{\rightarrow} \{P_1, \ldots P_n\}$$

Where:

P_i denotes the predicted states of R_i

G maps internal and environmental sets states.

For next states:

S next selects preferred future states

next-state = $S_{next} \rightarrow G \quad (e\ ,\ s)$.

For past states:

next-state = $G \quad (s_o, \ldots s_n)$ with G computable.

Applying G to a specific sequence of values $\langle e, s \rangle$ results in trajectory instance for every P_i in $\{P_1 \ldots P_n\}$.

Table 3 expands this logic using the original contributions by Ören and Yilmaz for agent-based perception representation [Ören and Yilmaz 2004]. It presents the anticipatory components used in HBR AP model V&V and incorporates the rudimentary principals of perception:

	Current Reality ($R_{i's}$)	
	Past or current state	*Predicted future state*
	$(s_o, \ldots s_n)$	(P_i)
Other objects, people or events	Perceived image of others and events	
Self (decision makers, supporters, followers, and/or events related to one's own object)	Perceived image of self and or events related to one's own object	Behavioral anticipation of others and events

Table 3. Categorization of Perception

A logic for AP depends on mapping mental processes and symbols for the critical navigation "world view" that is both internal and external to the perceiver. The perception of the external world begins with the senses, which leads to empirical concepts representing this "world view," within a mental framework that relates new perception concepts to pre-existing ones [Gibson 1966, 1979]. Consequently, the concept of AP in the model includes a paradigm of sensing, representation and knowledge organization prior to decision making for the complex and uncertain, simulated, critical navigation domain. This addresses the design and implementation of a reactive/cognitive AP architecture required for valid agent-simulation.

SME self-experience (empirical learning) agent behavior embodied in the agent software model is generated for each agent at run time in the simulation and results are automated decision-making responses. The automated agent responses are influenced, directly or indirectly, by pre-programmed perception variables (as identified by the SMEs) and constitute "artificial" selections or responses. The AP test instance assumes that the internal and external processes of a system share a purposive goal or accomplishment. Within this framework, all AP objects may dynamically participate. In the very simplest of forms, AP represents a higher order of sensory detection, interaction and interpretation resulting in an automated response that encompasses competition among simulated objects within an AB environment [Simonin and Gechter 2006].

This logic helps define the role of perception within SA and establishes meaningful approaches for introducing its representation in simulations through [Weixiong 2000]:

- Identifying the natural relationships between perception and SA
- Providing an ontology for the AB object interactive processes

- Providing logic to support optimization methods that ensure consistency in AP representation
- Providing repeatable V&V approaches for AP based on SA experiments for agent HBR

An agent simulation mirroring this logic incorporates the semantics used for the method introducing an AP HBR [Ghose and Koliadis 2007]. Agent perception τ represents a state space whose number and weight retain a potential to result in function, or functions, as depicted in Figure A-26 [Schematic Description of Agent Perception].

Where:

St = the set of states that agents of type t can assume;

Perception$\tau = \Sigma\tau \rightarrow [N\times Wf1] \ldots [N\times Wf|F|]$ is a function

Actions t = the set of actions that agents of type t can perform

Solutions for identifying this perception include the elicitation of a shared group task that requires specific individual response(s) based on identifiable elements of anticipation. Associated with the identification of perception elements are SME identification of visual cues and *gestalt*s which are recorded by ACTA knowledge audit worksheets. A meta-model is created that maps identifiable UML objects to a simulation. One objective is to provide an engineering object-oriented structure that is an accurate artificial representation of the specific perceptual task. Simulation tests then establish a baseline for measuring the artificial behavior. The collection and analysis of this information validates, or invalidates, a correlation between simulated perception and behavioral responses found in the actual experiments [Balci 1996].

Projections are that task analysis and critical decision methods provide credible perceptual input for an agent simulation. Further, an expectation is that experimental data validate the design of the AP method. A realistic reason for introducing AP to HBR is the provision of repeatable guidelines to validate a simulation that mimics the cognitive responses of the experiment based on anticipated and perceived action(s) of a future event. As shown in Figure 8, agent-objects representative of results from these cognitive experiments provide a measure for further validating HBR simulations beyond the simple use of "face validation" [Balci 2003].

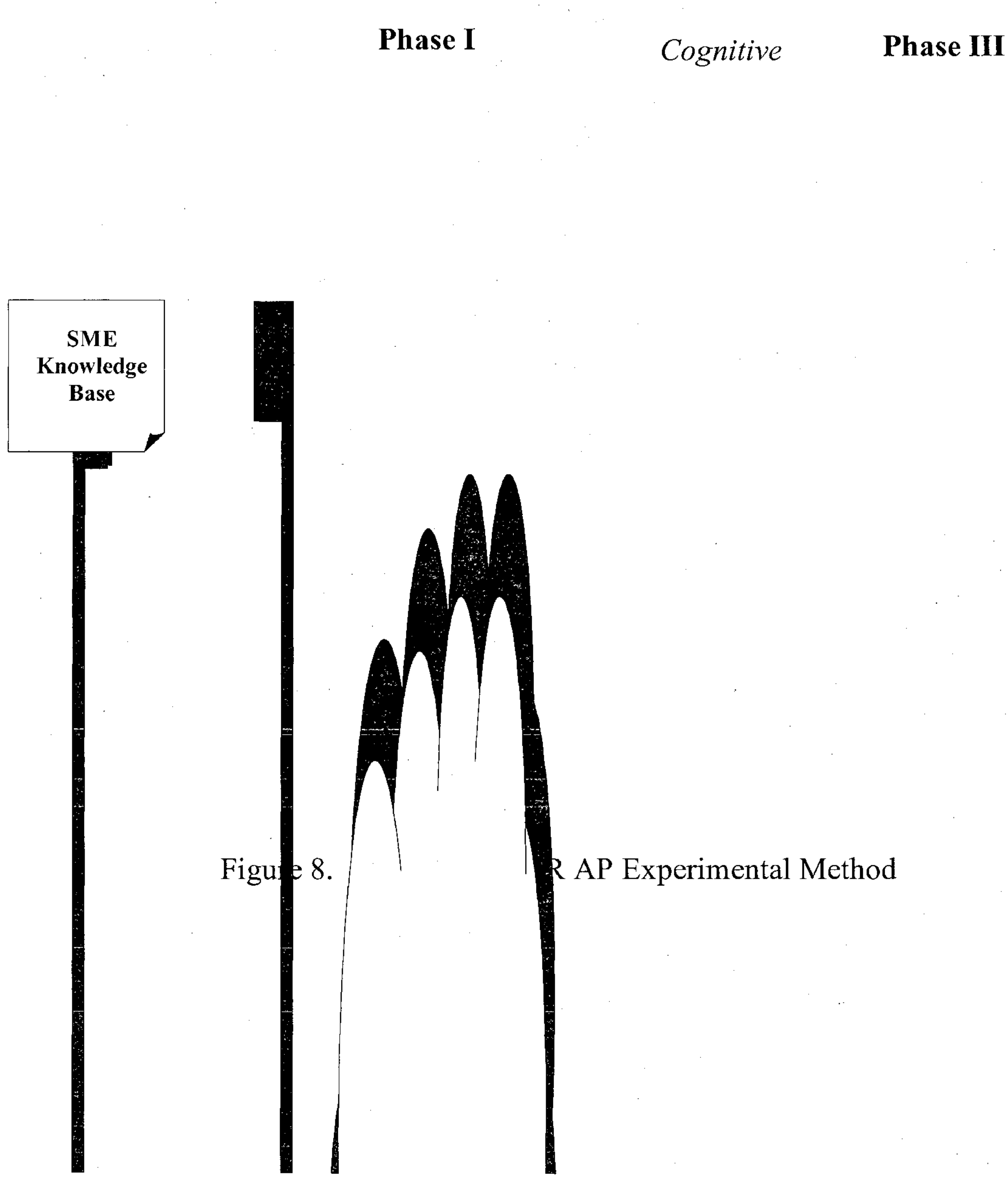


Figure 8. [illegible] AP Experimental Method

A generalized V&V approach used for introducing AP to improve HBR for agents includes:

Phase I: Scenario and conceptual approach validation

a. Intended use

b. Specific requirements

c. Referents

Phase II: Validated against SME Knowledge Base for the identified critical navigation tasks

Phase III: Cognitive representations developed

a. Experimental scenario objects

b. Test simulation agent-objects

c. Approach logic

Phase IV: Simulation developed using AnyLogic™ software and integrated into HBR of AP [AnyLogic™ 2006]

Phase V: Overall HBR verified and validated

a. Verified with SMEs

b. Validated with Analysis of Variance (ANOVA) simulation test results [Law and Kelton 2000]

The intent is to replicate perceived events using specific, critical, task-expert knowledge by agents as HBR for perception in a simulation. By conducting an experiment to test the influence of expert decision-making on simulated agent perception, any variance of a human decision with that of its agent should be detected.

SMEs directly interact with the HBR in gaming scenarios, enabling an elicitation of this information and assisting with the controlled development of the experimental model through pre-created scenarios using Quest 3D® gaming software [Quest 2006]. This allows them to observe the resulting behavior. Using this method, SMEs are able to determine whether the observed behavior meets the intended "use requirements" for realism [Dayton 1993].

A validating premise is that an applied, critical navigation instance of the AP method presents an accurate representation of reality [Xiufeng *et al.* 2004]. It presumes that SME knowledge for this reality (presented as a conceptual model) provides sufficient reference based on their inherent and specific skills, experience and expertise [Hoffman *et al.* 1998; Klein 1989]. Comparing this knowledge base with the actual HBR establishes a means of measuring anticipated behavior [Ören and Yilmaz 2004]. This measure is captured in the final tests.

It is important that acquired SME information effectively validate the method for introducing AP as a deliberation-action mechanism for a specific and intended use [Cassimatis *et al.* 2004]. It is also important to characterize agents as HBR through a set of possible actions that are simulated for the selection of an action to be undertaken based on the internal state and the position of the agents themselves [Riecke *et al.* 2002]. A position in the simulated environment defines the situation in which agents are acting. The situation relates to a potentially complex combination of internal and external events and states. The combination of both SME confirmation of the HBR events and states and ANOVA test measurements using a software replication of these events and states, increase the credibility of the AP method and further supports its accurate validation using expert knowledge [Hoffman *et al.* 1998].

As shown in Phase II of Figure 8, expert knowledge implies a dependable knowledge base usable for developing an UML engineering approach for HBR. Given that unconditional validation or “absolute truths” are not always attainable, accepted V&V techniques help reduce error [Sterman 2000]. Particularly, with the validation of cognitive models, many issues, as previously stated, make V&V difficult to accomplish. This is also true of ensuring uniform standards for test implementation [Uhrmacher 2000]. These additional considerations help reduce these limitations for an AP method validation:

- Use of ship navigation experts
- Observation of referent bias
- Documentation of expert perspective for model representations
- Measurement of model performance
- Accurate expert perception representation

As shown in Phase III of Figure 8, generalized applied methods for V&V use SMEs during ACTA to elicit and select the critical group tasks within a specific environment [Bonecito and Burns 2003]. Experimentation includes testing SMEs for behavioral responses triggered by visual perception, particularly visual cues, for these critical action scenarios [Yantis 1992]. SMEs are employed with a goal of revealing situations, objects, or observed responses that are used by agents in a simulation to represent this behavior.

In Phase IV of Figure 8, expert knowledge integrates with agent simulation providing an applied method for validating HBR for SME anticipated behavioral response(s). Observations seek a behavior correlation between agent perception responses similar to real-life events. The design of a V&V approach is to:

1. Develop a conceptual model as a mathematical, logical representation of a scene that mimics the real-life event. In this case, use a critical task scenario as an applied application to test the model [Sprague *et al.* 2007].
2. Validate the model application using SMEs and the techniques mentioned earlier.
3. Verify the logic associated with the simulation design, in this instance, that of the interaction of agent-objects [Bergmann, *et al.* 2003].
4. Build the representative agent simulation.
5. Perform simulation test runs and compare results with SME experimental field tests.
6. Make observations and document the test results.
7. Present findings as depicted in Phase V of Figure 8.

Precise perception, goal and action knowledge repositories are obtained supporting methods to introduce and improve AP in HBR. This information reinforces the logic used in the simulation. Using this information, the model also applies causal reasoning to determine influences on event state representation for the simulation[21] [Pearl 1989, 2000]. It ensures that the agent simulation processes are developed and structured

[21] e.g., perceived, goal and action states.

in a manner that is meaningful to the concept of perception within HBR [Oaksford and Chater 1994]. This knowledge approach manifests into a taxonomy outlining a meaningful hierarchical categorizations of V&V referents for the model that reflects natural relationships between SA and AP [Goldstein 2002].

3.3 Detailed Application

A simulation experiment introducing AP to improve HBR applies ACTA, SAGAT, SART and gaming scenarios to an agent, problem-solving approach. This decision process encompasses reactive agents operating in a goal-oriented system, e.g. safe navigation of a restricted waterway, allowing them to evolve over a given discrete time period and within a constrained state space [Riecke and Heyde 2002]. A simulation developed using g*estalt*s represents the critical task. Injecting the test data from SART gaming scores into the agent simulation, calculated anticipated responses result in an outward emergence of perception.[22] This surfacing of AP helps characterize definitions used to develop taxonomy of the method [Harmon 2002].

Empirical data from this experiment interprets emergent behavior of recurring interactions among agents within a constrained environment representing critical decision results. Evidence obtained from experimental applications of the AP method representing the defined human perception process helps validated HBR decision-making within the agent simulation. Simulation runs inject state-of-the-art Unmanned Surface Vehicle (USV) perception rule sets, with expert scores obtained from SART and agent simulation

[22] Anticipated responses should not conflict with decision theory. A purpose of this research was to provide a method to anticipate actions of a specific task resulting from a perceived mental model of the critical task environment.

scores for comparison. The application seeks to show that human, expert decision-making outperforms current state-of-the-art rule-sets when making critical decisions.

For the experiment, it is imperative to identify a method for binding human rationale with agent behavior to connect a cognitive behavior to an agent. The rationalization of cognitive behavior for HBR agents interacting in a critical scenario presents itself as excessively complex for absolute representation at the micro level. Therefore, to make a viable HBR assumption for a respective agent, the experimental method incorporates local and partial principles of rationality. New patterns of behavior emerge when modeled agents, whose actions with perception values obtained from cognitive task test results, face spatial constraints, thus allowing optimism for using this method of model execution [Holyoak and Simon 1999; Lowe 1987].

By using a method that introduces AP into HBR, it is possible to conduct experiments that mimic expert behavior based on visual cues. Using these visual cues presented by SMEs for goal-directed behavior within a hazardous environment, both cognitive experimental vignettes and an AP simulation may represent a critical event. This approach by comparing game-based, cognitive, field experiments using SMEs with observations of agent-based simulation test scores, tests the following hypothesis for an expert system:

H_1 = A relationship exists between agent AP simulation test results and SME cognitive experimental results.

From the results, the decision is to accept, modify or reject the hypothesis as stated.

Structured visual cues, knowledge repositories, logical processes and experimental organization provide methods for introducing AP as a means to improve HBR. These taxonomies are meaningful hierarchical categorizations of topics reflecting the natural relationships between SA and perception [Moya and Tolk 2007]. The taxonomies categorize a critical navigation scenario involving a target population that is engaged in repetitive, human performance behavior(s) within a shared group environment [Wickens *et al.* 2004]. Test results measure explicit behavior from the agent logic simulation. This developed agent simulation includes constrained environmental surroundings where participants encounter alternative decisions associated with the task of hazardous ship maneuvering [Finke *et al.* 2007; Janis and Mann 1977; Lowe 1987].

In developing the visual simulation for the experiment, cognitive test questions were bound to UML classes and instances demonstrated within these object hierarchies. This temporal-phase information addressed the SAGAT test question, "What are probabilities the vessel will turn left, right, maintain course or stop?" Referencing this approach within the context of formalism, specifications assume that events specific to the experimental scenario are deterministic in nature and apply to an agent HBR representation as follows:

Deterministic system 8-tuple $\langle T, U, Q, \Omega, S, \delta, Y, \lambda \rangle$;

T is a "time set" for a continuous system $T = \Re$ (reals) and for discrete time systems $T = \Im$ (integers);

U is the input set, e.g., HBR of AP object probabilities representative of anticipated turns;

Q is the total state set representing the HBR of an AP object;

Ω is the set of acceptable agent input functions d.

In this case, this contains a set of input functions that could arise during critical ship navigations. Often, due to physical limitations, Ω is a subset of the set of all possible input functions ($T \rightarrow U$). These acceptable functions are representative of scores obtained from ACTA.

S is a state set (product of all the finite state sets, e.g., agent value sets)

δ is the transition function as defined: δ: $S \times T \times \Omega \times \rightarrow S$

Y = the output set. Y is where we should observe for variances between U and Ω

λ = the output function: λ: $T \times S \rightarrow Y$.

The experimental scenarios and visual cues used for ACTA are not dependent on the absolute value of time; therefore, we may consider these scenarios time-invariant. This allows us to simplify the transition function δ:

$S \times T \times \Omega \times \rightarrow S$ where $\delta\,(s, t, \omega)$ results in a state from starting the scenario in s and applying input ω for a time duration of t units.

However, the time variance for agent HBR of AP represents simulation of agent response probabilities of a non-deterministic nature that have possible transition probabilities. Thus, response to visual cues varies with time and compares with methods found in MAS using the same "perception→ deliberation →action″ process. This is important to demonstrate that agent triggers, perceptions and states directly map into a hybrid DEVS formalism used to support HBR of AP. The use of a coupled model, where "perception→ deliberation → action" dynamics represent atomic models, shows a dependency between objects, their state space and corresponding, constrained, navigational environments [Giordano *et al.* 2004].

Agents represent atomic models and show state space dependencies with their relationships as represented by the DEVS hybrid model. The specification for the HBR AP model using the previous formalism considerations is as follows:

$\text{H - DEVS - CA} = \langle X_{self}, Y_{self}, D, \{M_i\}, \{I_i\}, \{Z_{ij}\} \rangle$

X_{self} = External input (consists of all the AP agent values)

Y_{self} = Output for all cells components in the hybrid DEVS model corresponding to St or AP agent state

D = C (Components in the DEVS model that correspond to AP state space)

$\{M_i\}$ = Atomic DEV components

$\{I_i\}$ = (Set of influences of an agent-object, - constructed by one of the AP agent perception triggers that contains "influencers")

$\{Z_{ij}\}$ = The *i, j* output-to-input translation

$\{s_i\}$ = The system *i,* output-to-input translation

$\{Z_{ij}\}$ converts the output of the AP agent's value into a tuple containing that value and the offset between an agent and another navigational object:

$$Z_{ij} : Y_i \rightarrow \quad X_j$$
$$s_i \rightarrow (s_i, i - _j).$$

The output-to-output translation translates the output of each AP agent into a tuple:

$$Z_{i, self} : Y_i \rightarrow \quad Y_{self}$$
$$s_i \rightarrow (\ldots, s_i, \ldots).$$

Objects map onto atomic DEVS components:

$$M_i = \langle S_i, ta_i, d_{int, i}, X_i, d_{ext}, d_{confl, i}, Y_i, \lambda_i, \rangle, \forall_i \varepsilon D.$$

Values of the components are those from the AP agent-object value set ($Si = V$) with the time advance (*ta*) set to the same arbitrary non-zero value Δ. This allows for object synchronization: $ta_i (s_i) = \Delta$. The component's state is not changed by the internal transition function, $d_{int} (s_i) = si$.

A generated set containing the agent-object value by the output function:

$$\lambda(s_i) = \{s_i\}, \text{ where } s_i \varepsilon Y_i = V.$$

Other than the influence of the *gestalt* cue triggers, time, internal network functions and random number generator there are no global external input into the HBR of AP; therefore, there is no external transition (d_{ext}) Due to a synchronous operation, only the confluent transition functions are used:

$$d_{confl\,i}(s_i,\ e_i,)\quad x_i^b\ = {}_1(v_i)$$

Where: v_i is a vector with the same dimensions as the AP Agent-object (AO) vector. e.g., HBR AP triggers:

$$v_i[t]\ =\ s_i,\ \text{for}\quad t\ \text{which}\ AO[\,t\,]=0$$

$$v_i[\text{proj}_{\text{offset}}(x)\,]\ =\ \text{proj}_{\text{value}}(x),\ \forall\, x\,\varepsilon\, x_i^b.$$

AO values and offsets transmit via messages from the current vector:

$$x_i^b\ = \{(\text{v, offset})\,|\,\text{v}\ \varepsilon\ \text{V, offset}\ \varepsilon\ \text{C}\}.$$

The agent simulation tool AnyLogic™ allows for construction of the navigator reactive agents for the ANOVA tests [AnyLogic™ 2006]. Agent specifications for the simulation include populations that share time, space, network and communication variables [Borrie *et al.* 2007]. In addition, the simulation test software for the environment contains object-oriented software code classes, properties and variables that reflect the taxonomy of the UML model [Booch 2003]. These classes represent visual object *gestalt*s and cues identified in the ACTA used for SAGAT/SART gaming experimental vignettes, previously discussed [Bosse and Jonker 2006].

Through model constraints, "own ships" objects have equal influence with "perception" class objects [Lowe 1987]. "Own ships" represent scalar vectors in the simulation and contain properties of ship contacts and their approaching states. They are included in the model as representative of actions identified in the ACTA.

For clarity, the string parameter PopulationName "Model" defines the population of an agent for the encapsulated AP agent-objects "Ownship Perception" and "Ship Approaching" (Appendix, Figure A-10). The population retains its global parameters from the settings of the AP agent parameters respectively. Initialization of network and layout within the population occur on a special event scheduled at time 0. All project objects programmed create at model run-time as follows:

- The DefaultNetwork is of ALL IN RANGE type and is constructed.
- The default continuous layout is applied.

The first step ("tick") of the model occurs immediately following initialization. A startup code for "Ownship" agent-object is listed in Table 4:

Startup code	model = (Model)getOwner();
	//animation
	x = radius*cos(angle);
	y = radius*sin(angle);

Additional class code	//getOwner()
	Model;
	
	//animation
	double radius = uniform(25);
	double angle = uniform(
	2*Math.PI);

Table 4. Time Start Code for Ownship Agent Based Object

The UML is animated using software parameters to both visualize the critical scenario and observe reactive agents gathering data on their emergent behavior [Meyer and Szirbik 2007]. This emergent behavior records each of the agent-decision variables used for ANOVA between agent responses and state-of-the-art rule sets. Collision avoidance and successful navigation in this experiment references robotic and USV rule-set algorithms for obstacle avoidance used to predict a collision between a moving point and a moving contact radius. Geometry in Figure 9 shows the collision area between these points. If a point and a circle of radius R are moving with constant velocities such that they satisfy the following algorithm at any given instant "in time" then they will continue to satisfy the equation $r^2 V_\theta^2 \leq R^2(V_r^2 + V_\theta^2)$ for future time [Chakravarthy and Ghose 1998].

Where *R* is the radius of the circle, *r* is the distance between the point and the center of the circle, and Vr and V_θ are the relative velocity the length of and at right angles to the line connecting the two objects. When trying to predict a collision, the following cases apply:

Case 1: $V_\theta = 0$ and $Vr < 0$ (the objects are on a straight line collision course with each other)

Case 2: $V_\theta \neq 0$ and $Vr < 0$ (a collision is possible but conditions must be discovered that could result in a collision course)

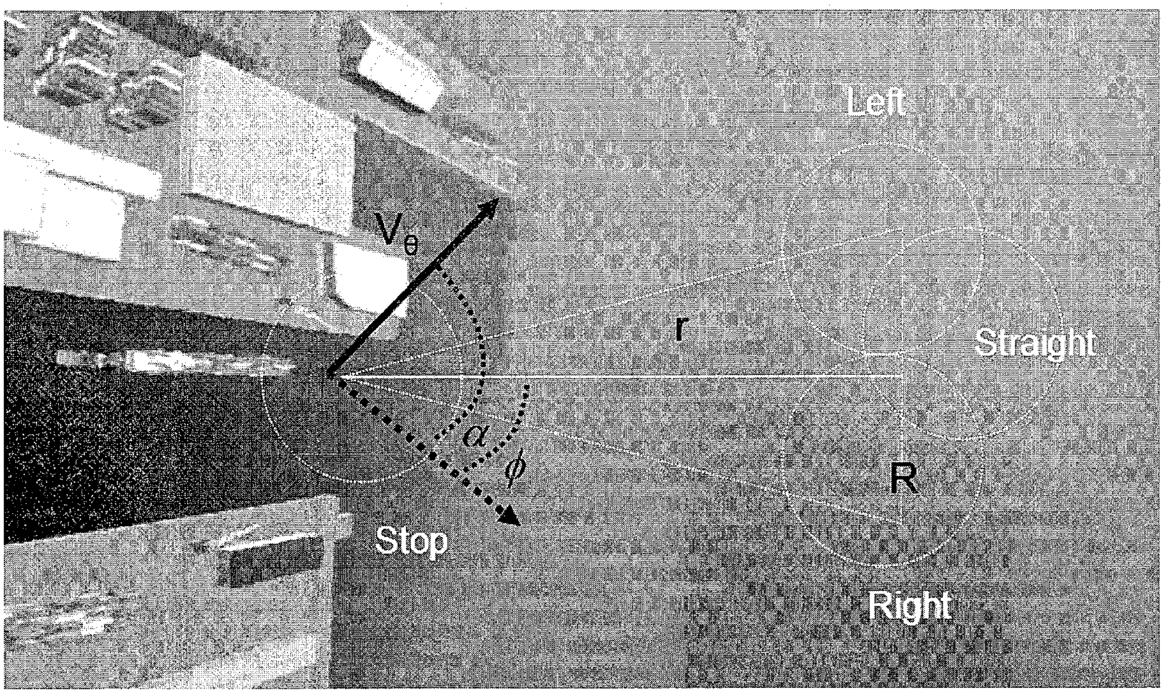


Figure 9. Simulation Animation and Approaching State Environmental Constraints

Simulation time makes use of the AnyLogic™ AgentBase function: CONTINUOUS. Using this function, a chart timer, state-chart, and equations independently control agent time. As shown in Table 5, these generate individual AP agent activities over time.

ChartTimer	ShipArrivals
Timeout	exponential(ContactRate)
Expire At Startup	No

Expiry Action	//generate Ownship
	Ownship c = new Ownship();
	setup_Ownships(c, NContacts);
	
	// attach a Ship - !!!
	for(int i=0; i<perception.size(); i++) {
	  Ship s = perception.item(i);
	  if(s.Ownships.size() < s.OwnshipsMax)
	  {
	    s.Ownships.add(c);
	    c.Ship = s;
	    break;
	  }
	}

Variable	showperceptionLinks

Table 5. Chart Timer for Ownship Agent Based Object

The CONTINUOUS function implements the spatial variables for the agents. Each agent has a pair of real coordinates (X, Y) and there are no initial run-time restrictions on the density or location of the agents. Specifying AnyLogic™ Xdynamic and Ydynamic parameters control location. Entry and exit actions resulting from transition algorithms control the state space (Appendix, Tables A-15 and A-16) Xdynamic and Ydynamic Algorithmic Functions). Programmed AP agent animation initially positions UML objects on the animation template according to their initial coordinate values.

Representing the critical scenario environment, boundaries limit agents (representing expert navigators) to 30-degree "cone of collision" coordinates, allowing for reproduction of a realistic, partially-observed state space [Chakravarthy and Ghose 1998]. Agent interaction within these coordinates become important factors for linking UML visual cues from the SME cognitive tests to the agent simulation [Riecke *et al.* 2005]. Within this state space, visual cues help define what constitutes a critical decision [Thorsden 1991].

Expert-defined *gestalt*s, visual cues and their triggers provide the central foundation for building UML objects and validating the critical, decision-making gaming vignettes for experimentation. These become the template for creating the simulation objects where reactive agents, given controlled input variables, are able to represent either a "novice" or "expert" response to the critical scenario given H_1 [Hoffman *et al.* 1998]. Agent output variables represent decisions from simulated emergent behaviors. Observations seek variances between these reactive agent emergent behaviors with the decision results obtained from SAGAT/SART gamming vignettes, where SMEs score their anticipated action.

Three UML models represent the agent state transitions. In the first model, cognitive elements from anticipated scenario collision avoidance transitions discovered in the critical navigation action hierarchy, map to UML initial agent perception and state transition objects (Appendix, Figure A-4). As shown in Figure 10, cognitive mapping provides input model ($M_1 \ldots M_{10}$) parameters determining the initial contact and perception states. These states augment the two remaining transitions; *gestalt* cue state transitions and anticipated response transitions shown in Figures 10 and 11 respectively.

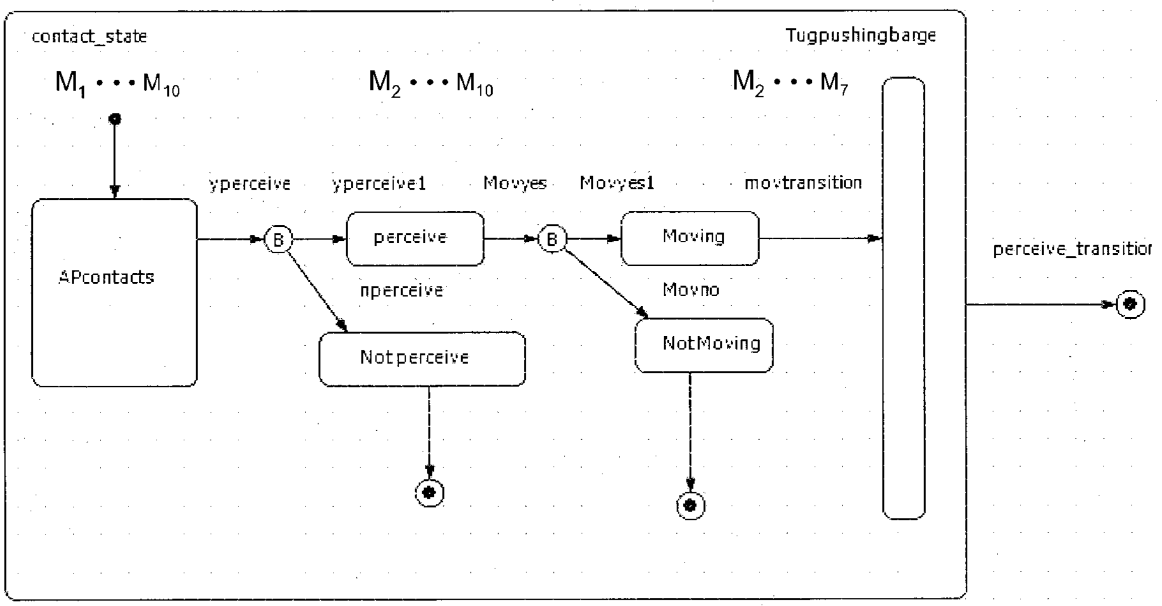


Figure 10. Cognitive Mapping (Input $M_{1 \ldots 10}$) to UML of Initial Agent Perception and State Transition

In the second UML model, expert-recognized cues present an anticipated object behavior and transform into usable *gestalt*s for a cue transition state (Figure 11). Cognitive mapping provides input model ($M_1 \ldots M_7$) parameters determining the initial contact and cue transition states. This appropriately provides a discrete representation for cues used in a simulated, stochastic agent process. Agent simulation applies these cue derivatives for object recognition within the scenario (Appendix, Figure A-4).

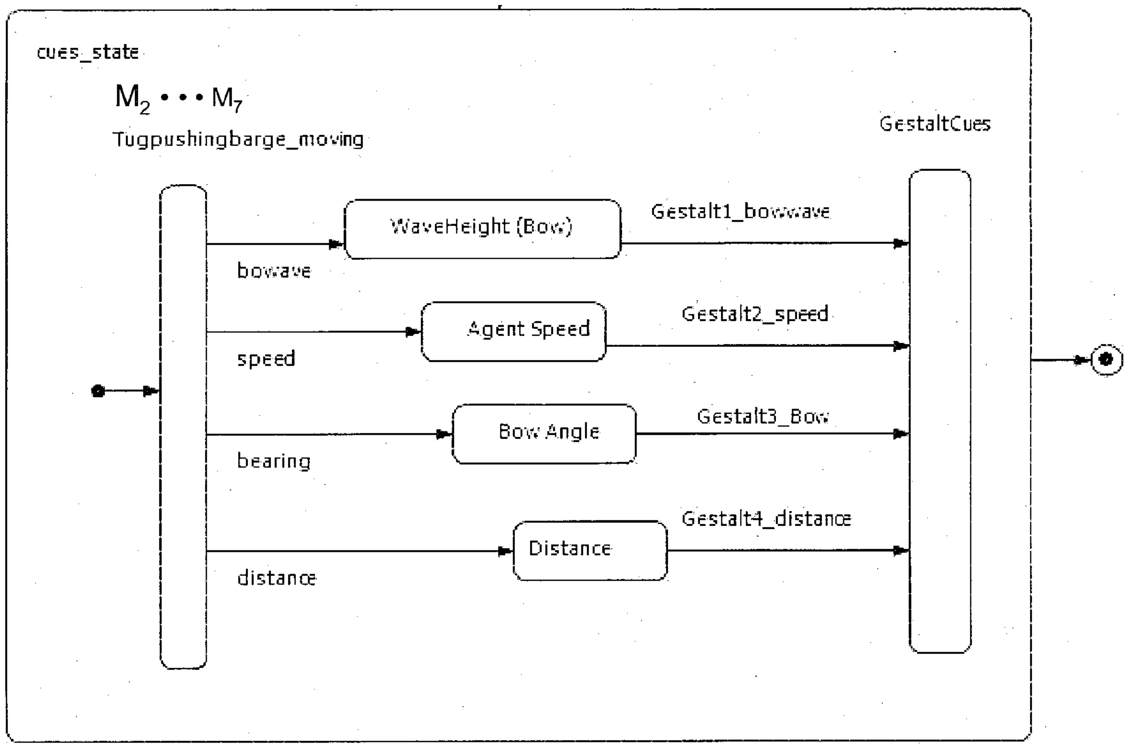


Figure 11. Cognitive Mapping (Input $M_{1}\ldots_{7}$) to UML Cue State Transitions

In the agent simulation, decision variables for stop, left, right and straight implement through transition vectors. As shown in the third UML model, the agent has "free play" within all three of the state spaces influenced by all state transitions. Figure 12 illustrates these "triggers" (state transitions) as vectors in the UML model. The first trigger is a timeout function and is a cyclic timeout exponential (contact rate) for the object "ShipArrivals." In this case, transitions executed after the <statechart> have been in the current state for a specified period. The simulation time used for the navigation model is in real-time mode (specified speed) in model time units of 1 per second.

The second trigger uses a signal order where the transition is an object (event) in the statechart queue matching the programmed description <statechart>.fireEvent()

function for inserting a perception event in the <statechart> queue. In the HBR AP agent simulation, these are:

<All Stop>.fireEvent (Immediately)

<Right>.fireEvent (Immediately)

<Left>.fireEvent (Immediately)

<Go Straight>.fireEvent (Immediately)

Otherwise, unless an event declared as <deferred>, is discarded.

The third trigger uses specified conditions that are true. If the object remains a continuous variable, the conditions are constantly checked. If not, the object can be acquired and checked through a setModified() command. When there is no trigger, no event will occur. If a trigger appears for agent-objects, the function <guard>.randomTrue is checked. If <guard> is false then the transition is not taken. If <guard> is true a transition is taken, and the function <action> is executed. The agent-object <guard> function is a Boolean expression and <action> executes via java code. A state diagram (statechart) defines the periodic sequence of events. Event execution is atomic and the actions associated with an event are uninterruptible. Triggers will only occur when the agent meets all parameters of the three state transitions. Results displayed in the simulation are uniform, random variates controlled by (dx/dt) and input parameters from the three state transitions (Figures 10 - 12).

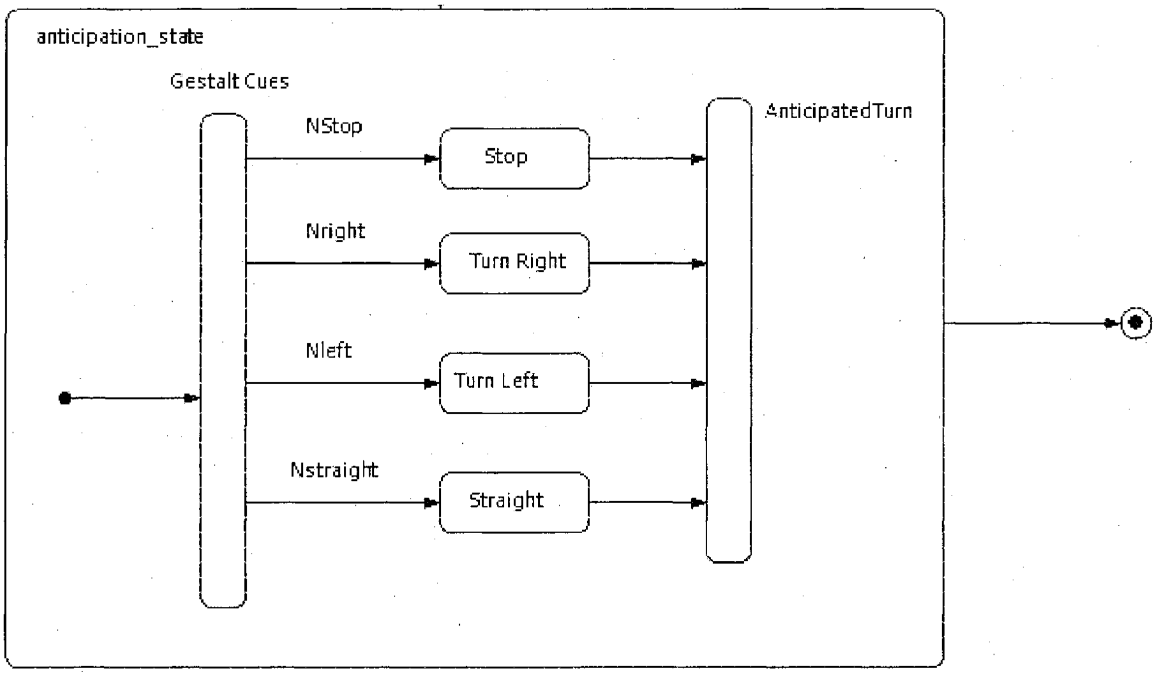


Figure 12. UML Agent Model Showing Anticipated Decision State Transitions

Using a meta-model, each agent is able to maintain contact within the same population within the UML organization [Zeigler 2000, Appendix, Figure A-5]. The agent command getContacts() returns a vector of agent-objects connected. A predefined network type using the AnyLogic™ RANDOM command allows the agents to connect. Agent animation defines the AP agent command AgentBase allowing for the display of links between the connected agents as straight lines.

It is imperative to ensure that the simulation used to conduct the test has both consistency and can provide reliable results [Zabinsky *et al.*1992]. Parameter, variation, simulation runs for environmental values are tested. Each value tests at 100,000 iterations with a contact rate of 100 per second. Initial observations show noticeable variations of model behavior at iterations under 50K runs. Stabilization occurs near the

100,000 iteration. The generated input variables do not test the inherent, Java random number generator; however, these are usable as intended for testing object interaction within the simulation model. Parameter variations are shown for HBR AP model generated objects (Appendix, Figures A-20 and A-21).

Observations of simulated agents using human expert values demonstrate a similar variation of model behavior at iterations under 50K runs. However, expectations of the variation runs for agent decision values remained consistent with expected values for initial simulation contact rates. This is anticipated due to the root agent-objects for "right" and "straight," each having higher values from SME cognitive test scores. The agent for "straight" did not show a variance as large as anticipated compared to the variance observed for "turn right" with similar values, but did retain higher values than "turn left" and "stop" (Appendix, A-20 and A-21).

Optimization tests verify that optimum agent values are presented in the simulation. A decision to use either independent algorithms or optimization methods found within the agent software needed to be resolved, as the agent simulation is context dependent [Mitra 1998]. Normally, complex systems are limited to problems formulated and programmed as mathematical models of linear, nonlinear and integer types with problem specific heuristics generally developed on a case-by-case basis. The AnyLogic™ optimization program was chosen. It has an integral Application Program Interface (API) that uses the two reliable population-based meta-heuristics AutoCore™ genetic algorithms and scatter search [Glover *et al.* 2003].

AutoCore™ API inherent in AnyLogic™ provides a non-deterministic learning

AI engine to support simulation optimization [OptQuest™ 2006]. This API supports the OptQuest™ optimizer integral to the simulation. This optimizer uses heuristics, neural networks, and mathematical optimization methods to find the values of model parameters that give maximum or minimum value of the objective function. It provides optimization for the simulation with constraints and tests under uncertainty.

Model simulation optimization runs used the parameter "NumberOfContacts" for each of the continuous agent-object function associated variables Nleft, Nright, Nstraight and Nstop. Parameter Optimization provided the best value for input variables to maximize the parameter "NumberOfContacts." These considerations are acceptable and the chosen optimization results are included into the AP simulation.[23]

Expert ACTA, SAGAT and SART experiments contribute to a perception meta-model capable of replicating visually cued responses (Appendix, Figure A-5). These techniques elicit working knowledge of Subject Matter Experts (SMEs) for a navigation critical task to correlate with an applied engineering use case [Booch 2003]. Software engineering methods extract information gathered, using these techniques and allowing for construction of both agents representing experts and UML objects associated with the meta-model.

Agents use values derived from this knowledge elicitation as input for influencing their behavior within the simulation. A hypothesis validates the introduction of AP as an agent representing expert decisions. Agents acting as experts are simulated and the results compared with state-of-the-art USV rule sets. The assumption is that an expert system

[23] A major objective of optimization is to ensure that the HBR software platform provides reliable, consistent and repeatable performance.

will perform better at collision avoidance than a rule set and there should be a distinct variance between resulting, decision output responses.

Techniques presented in the experiments are complementary and designed with the purpose of extracting different aspects of cognitive skills [Bacchus *et al.* 1993]. First, SMEs brainstorm and define a task they feel is critical in nature requiring expert decision- making that is not available to a novice. Safely maneuvering a vessel to avoid collision in a restricted waterway is the chosen task. Based on the nature of the task, SMEs complete a questionnaire ensuring they have the requisite expertise, skills and knowledge to successfully perform the ACTA as experts. (Appendix, Figure A-1)

Second, a knowledge audit worksheet provides the survey of expertise issues required to complete the specific critical ship navigation task and subtasks. (Appendix, Figure A-3) Each aspect of expertise identified is: [24]

1. Analyzed to identify specific examples within the context of a critical task.
2. Observed for cues and strategies used.
3. Determined why it presents a challenge for those who were not experienced ship handlers.

This allows for an exploration and initial screen of difficult cognitive items related to the ship navigational task of a tug pushing a container barge in a restricted waterway using the comprehensive interview techniques [Klein 96].

---

[24] The knowledge audit assists with identifying ways in which expertise uses a ship-handling domain and provides examples based on actual experience. The knowledge audit defines and organizes knowledge categories that characterize critical ship navigation expertise.

After conducting the knowledge audit probe, the experts compiled cues and strategies as well as an explanation of difficulty associated with the critical navigational scene (Appendix Table A-2). This presents a broad overview of perceptual skills needed to safely complete the navigational task.

The next step involves a simulation interview identifying the critical events, actions and assessments associated with the SME cognitive processes within contexts of the critical navigational scenario (Appendix, Table A-4). In the simulation interview, critical cues begin to emerge. The use of a gaming vignette, as previously discussed, helps provide a catalyst to obtain the cognitive data that is difficult to obtain via the other interview techniques [Camerer 2003]. This allows for additional probing, to include:

1. Situation Assessment [Taylor 1989]
2. How situation assessment impacts a critical decision
3. Identification of potential errors that a novice would be likely make in the same scenario

The simulation interview allows for a better understanding of the subjects' cognitive processes within the context of the critical incident.

Upon success with the previous interviews, a comprehensive knowledge audit is conducted (Appendix, Figure A-2 and Table A-3). Using worksheets, probes provide a sampling of the comprehensive challenges associated with the scenario of a tug pushing a container ship in a restricted waterway (Appendix, Table A-1).

A critical, navigational, action hierarchy graphs the cognitive process for identifying objects recognized by order of the perception associated with the critical navigation and task of safely avoiding collisions within a restricted channel (Appendix, Figure A-4). This cognitive process maps to UML objects for building both the gaming vignette used for SAGAT and the agent simulation. From these objects, a meta-model applies software engineering methods to cognitive process.

Initial experimental sessions defining the critical task took a period of approximately two weeks to complete. Each member was allowed time to reflect upon their ship-handling experience prior to collaboration, ultimately agreeing on the task to use for scenario development. A consensual critical scenario is developed into ".cgr" files using Quest 3D® Virtual Reality (VR) software and displayed as a vignette in a run time of version of the gaming software VSTEP Ship Simulator [VSTEP 2006; Quest 2006]. SMEs validate this finished critical navigation task vignette for accuracy and ensure it replicates the anticipated movement of an oncoming vessel.

Using the completed game vignette, participants rate their responses on a Likert (7-point) scale to reflect awareness of an evolving situation [Likert 1938]. Response options range from anticipation of whether the container barge will turn right, turn left, remain on course or stop. The test scenario employs a gaming format representing the critical task of observing the container ship and SAGAT, combined with SART methods freezing the scenario at intervals to score the anticipated responses.[25] These scores

[25] The simulation used for SAGAT developed in Quest 3D software, the visual object based on the results of cues, and critical objects identified in the ACTA and by SME face validation of the experimental scenario [Balci 1996].

represent expert decisions and become input variables representing anticipated container barge turns in the completed agent simulation. Using AnyLogic™, agent simulation runs are completed, mimicking the visual objects, *gestalts* and cues from the meta-model, as identified in the ACTA and game vignettes.

Overall, there were 22 SME participants in the experiment. Six of the 22 participants helped define and validate the critical scenario. Sixteen remaining participants observed the completed scenario vignette and scored the anticipated ship maneuvering results using SAGAT/SART techniques.[26] Each SME had an average of over 16 years of ship-handling experience, including hazardous navigation in restricted waterways. The SME qualification questionnaire documents SME professional ship-handling experiences (Appendix, Figure A-1).

[26] Each SME had an average of over 16 years ship-handling experience including hazardous navigation in restricted waterways

4. EXPERIMENT AND EVALUATION

If emergent, reactive, agent behaviors significantly differ in simulation runs from decision output parameters from SAGAT gaming (Likert responses), then "expert knowledge," as projected by SMEs, demonstrate an influence on decision results that are biased towards an expert view of events [Tanaka 2005]. If the emergent behavior of an agent decision does not significantly vary, then expert influence is insignificant. This would result in a conflicted agent perception of future events [Janis and Mann 1977; Lowe 1987]. The following experimental approach tests for this variance:

1. Obtain a sample size[27] and then employ a normal distribution to estimate the sampling errors.
2. With a normal distribution, large errors have a low probability and small errors have a high probability.
3. If an agent is simulated without using SAGAT/SART scores (referred to as agent B) and then simulated again with values obtained from SAGAT/SART (referred to as agent A), then it is hypothesized that agent A will perform in a manner more representative of the experienced knowledge obtained in the experiment than agent B. This test estimates the mean performance of A and B using a sample size *n* of trials for each.

[27] Sample size = 16.

4. A conclusion could be that A is indeed performing in a different way than B if its sample mean is sufficiently greater than the sample mean of Bs performance (values of n and the variance taken into account).
5. If the performance of A is not sufficiently greater than that of B, based on the samples, then the inference is that A is performing similarly to B in these samples and this is likely due to random chance.
6. Otherwise, the comparison concludes as statistically significant.
7. The larger the number n of samples and the lower the variance then the greater the confidence is that it is not due to chance.

To prove that the AP agent is an expert, meta-model representation and not a direct replication of expert input variables, trials must demonstrate a variance between the expert trials and the AP agent trials. ANOVA is used for these observations.

ANOVA observes heterogeneity between the means of agents defined as (A) and (B) [Cheng 1997]. A factor used in the test is the value of Agent (A) input variables for the anticipated SME action of an oncoming container barge and Agent (B) representing the behavior of the simulated perception agent. A t-test, based on the standard error of the difference between two means, tests the variance between the two means. Significance tests determine if an observed value of the measured statistic differ enough from a hypothesized value of a parameter to draw the inference that the hypothesized value of the parameter is not the true value [Gigerenzer, G., and D. J. Murray 1987].

The significance test calculates the probability of obtaining a statistic as different or more different from the null hypothesis (given that the null hypothesis is correct) than the statistic obtained in the sample. Procedures used for establishing the significance level include first determining the relation between the results of the experiment and H_0. Assuming that H_0 is true, the probability of a difference as large or larger is computed. Finally, this probability compares the significance level. If the probability proves less than or equal to the significance level, then H_0 can be rejected and the outcome can be determined as statistically significant. This experiment used a significance level of .05 α. The observation for significance testing errors includes:

(1) A true null hypothesis can be incorrectly rejected.

(2) A false null hypothesis can fail to be rejected.

These are Type I and Type II errors, respectively.

Assumptions are:

1. xA is the average of sample A, and s A is the standard deviation of sample A;

   H_1 = A significant relationship does not exist between agent simulation test results and SME cognitive experimental results

   $H_1 = A > B$, then $H_0 = A <= B$.

2. a will be based on the confidence level for the conclusions (e.g., to be 95% confident of the conclusion - reject the null hypothesis - then a = 0.05).
3. Set degrees of freedom = $n_A + n_B - 2$.
4. The overall variance $s^2 = (n_A - 1)\, s_A^2 + (n_B - 1)\, s_B^2)$ / d.o.f.
5. Find t_a based on table lookup of a and d.o.f.

6. Calculate $t = (x_A - x_B) / \quad s^2 * (1/n_A + 1/n_B)$, where n_A and n_B are then sizes of the samples (i.e., the number of test trials for Agent A and Agent B).
7. Accept H_1 if $t > t_a$; else accept H_0 or reserve judgment for another test.
8. Reject the H_0 and accept H_1 if $t > t_a$; or else accept H_0 or again - reserve judgment.
9. Both tests assume equal variances for the performances of Agents A and B. If variances are unequal, then one needs to adjust the d.o.f. for the test based –in this method - on the F-statistic, etc.
10. Tests assume i.i.d. samples, e.g., each trial is independent, and the development of Agent B was independent of the development of Agent A.

H_0 analysis of variance tested for equal population means that a direct relation exists between agent-simulated results and simulation results using expert, input variables from SME cognitive test scores:

$$H_0: \mu 1 = \mu 2 = ... = \mu a$$

The perception agent and human evaluation compare two estimates of variance (s^2)[28] for the agent value and the agent generation. Based on the variances within the samples, the Mean Square Error (MSE) estimate tests whether or not the null hypothesis is true. The second estimate Mean Square Between (MSB) is the variance of the sample means. The MSB is an estimate of s^2 should the null hypothesis be true. If the null hypothesis proves false then MSB estimates are something larger than s^2. The logic for the null hypothesis is as follows:

[28] s^2 = the variance within each of the treatment populations.

- If the null hypothesis is true, then MSE and MSB should approximate the same since both are estimates of the same quantity (s 2);

However:

- If the null hypothesis is false then MSB can be expected to be larger than the MSE since the MSB is estimating a quantity larger then s^2.

Therefore:

- If the MSB is sufficiently larger than the MSE, the null hypothesis can be rejected. If the MSB is not sufficiently larger than the MSE then the null hypothesis cannot be rejected.

Comparisons are conducted for the two estimates of variance using agent simulation. These test results yield observed variances supporting H_1. UML objects are used to represent expert decisions in the simulations. Appendix A provides a record of these comparisons, test results and observations.

The analysis and results support a central premise that a main effect of anticipated response, based on expert cognitive skills within a critical scenario, may be observed using agent simulation. The test platform analyzes how the distribution of a continuous Y variable (Agent A - the simulated agents) differs across groups defined by a categorical X (Agent B - agents with a value obtained from the ACTA scores). Group means are calculated and tested. This procedure allows for performance observations to include a best-fit model and [Law and Kelton 2000]:

- Analysis of variance to fit means and to test equality

- Nonparametric tests
- Test for homogeneity of variance
- Comparison tests on means
- Power details.

Plots render the true distribution of values as listed in Appendix A. By observing these plots, a general conclusion is that Agent B is performing better than Agent A.[29] (The values of n and the variance are taken into account.[30]) If, based on the samples, the performance of B is not sufficiently greater than that of A, then it is evident that B is outperforming A, thus, it is likely due to random chance.[31]. Otherwise, the comparison is statistically significant. Analysis confirms that test results are in support of H_1.

Agents UML objects simulate decisions based on perception of a competing object's movement within the constrained environment of a critical event scenario [Rational 2003]. Results show that SME Likert score estimates, using SAGAT/SART measures, differ for randomly, generated agent-objects than for agent-objects using expert scores as input variables. This contributes to the method introducing "perception" to HBR for improving expert influence in agent simulations. This is a measure to ensure that agent behavior is not conflicted [Lowe 1987].

Experts estimate anticipated ship object movement while observing a critical scenario and record a value on a scale of 0 (least likely) to 7 (most likely). This data

---

[29] If the Agent B sample mean is sufficiently greater than the sample mean of Agent A's performance.

[30] The larger the number of samples and the lower the variance, the greater the confidence that it is not due to chance.

[31] Agent A performance is defined as a manner more representative of the expert knowledge obtained in the experiment than Agent B.

analyzes a two (agent) by four (perception) ANOVA. Table 5 lists the simulation run data (AnyLogic™ Agent Software, version 5.3.1, build 758):

Contact Rate	1,000 per second
Number of Contacts	100
Model Time units per second	100
Total Simulation time	500 seconds
Number of Iterations	360,000
Independent Simulation Runs	16

Table 6. Simulation-Run Data for Agents

A paired t-test and parameter estimates compare mean Agent results. This allows the comparison of the overall differential mean scores test yielding a test statistic and probability for supporting the hypothesis findings. The second measurement analyzes variances for both Agents.

Figure 13 shows a quartile plot of SAGAT/SART agent expert choices and random agent response. Table 6 shows the results of agent response by choice, Tables 7 through 10 are result mean comparisons and Table 11 demonstrates the power of the test.

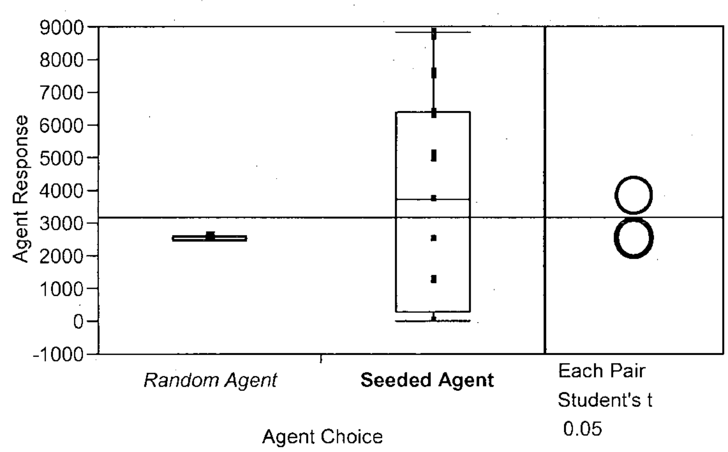


Figure 13. Quartile Plot of Expert Agent Response

Level	Minimum	10%	25%	Median	75%	90%	Maximum
Agent	2449	2449	2505	2524	2578.25	2590	2590
Expert	0	0	295	3700.5	6395.75	8706	8819

Table 6. Quartile Results of Agent Response by Choice

Difference	1320.80	t Ratio	3.260023
Std Err Dif	405.15	DF	63.03158
Upper CL Dif	2130.42	Prob>[t]	0.0018
Lower CL Dir	511.18	Prob>t	0.0009
Confidence	0.95	Prob<t	0.9991

Table 7. "t" Test Agent and Expert Results Assuming Unequal Variances

t	Alpha
1.97897	0.05

Dif=Mean[i]-Mean[i]	Expert Agent	Agent
Expert Agent	0.0	1320.8
Agent	-1320.8	0.0

Abs(Dif)-LSD	Agent	Random Agent
Expert Agent	-801.78	519.02
Agent	519.02	-801.78

Table 8. "t" Means Comparisons

* Positive values show pairs of means that are significantly different.

Level	Mean
Expert Agent A	3843.4219
Agent B	2522.6250

Table 9. "t" Test Means Level Comparisons

* Levels not connected by the same letter are significantly different.

Difference	1320.80	T Ratio	3.260023
Std Err Dif	405.15	DF	126
Upper CL Dif	2122.58	Prob>[t]	0.0014
Lower CL Dif	519.02	Prob>t	0.0007
Confidence	0.95	Prob<t	0.9993

Table 10. Results of "t" Test Comparisons for Expert Agent with Random Agent

a	s	d	Number	Power
0.0500	2291.87	660.39	128	.89

Table 11. Power of the Test

"t" Test Results Analysis:

The difference between the sample mean = 1320.80

The standard error of that difference estimate = 405.15

The ratio of this difference to its standard error; t-ratio = 3.260023

The degrees of freedom adjusted for unequal variances; DF = 63.03158.[32]

Probability:

The probability values (p-values) reveal the likelihood of getting the computed t-ratio again when, in the agent grouping, there is no difference in mean response between the groups.

Prob > |t| = 0.0014 (two-sided t test)

[32] Degrees of freedom = sum of the sample sizes less the number of groups.

The probability (P value) of obtaining the absolute value of the t-ratio (3.260023) is 0.014. This means that, by chance alone, there are only about 1.4 chances in 100 similar samples of observing a t-ratio larger than 3.260023 or smaller than -3.260023.

Prob > t = 0. 0007 (one-sided t test)

The probability of obtaining a t-ratio of 3.260023 or greater is 0.0007. There are only about 7 chances in 1000 similar samples of seeing a greater difference in-group means, which then gives a greater t-ratio.

Prob < t = 0.9993 (one-sided t test)

The probability of obtaining a t-ratio of 3.260023 or less is 0.9993. This means that about 99 times in 100 similar samples expect to see a smaller difference in-group means, giving a smaller t value.

Upper CL Dif = 2122.58

Lower CL Dif = 519.02

Confidence = 0.95

The upper and lower confidence limits are computed so that 95% of the ranges from similar samples would contain the true difference. If the group means observed were truly the same as in the population, then the t-ratio would have a t distribution with a mean equaling zero. Specifically stated, the greater the difference between the t-ratio and zero; the more unlikely the sample. Table 12 shows the combined agent responses.

Anticipation	SME Agent	Agent	Mean
Right	5223.50	2543.00	3883.25
Left	2919.75	2590.00	2754.88
Straight	4849.18	2505.00	3677.09
Stop	4849.18	2449.00	3649.09
Mean	3843.42188	2522.625	3491.077

Table 12. Combined Agent Reponses

Effects on the simulation are observed from expert-decision input variables. Experts who perceive and select a choice for anticipated container barge movement, positively influence agent-simulated responses. There is a mean difference of 1320.797 between the Expert Agent and Random Agent in the simulation results. This variation concludes that the means are different in the agent tested population, and further concludes that the sampling is not given to chance, confirming the SME agent as a representation of an expert system.[33]

Observation of a variance in μ Agent = 3843.4219 versus μ $\text{Agent}_{\text{random}}$ = 2522.6250 supports rejection of H_0. The fit Y by X data in Table 12 for the analysis used SAS-JMP, software version 6.0 [SAS© 2007]. Table 13 shows the mean and standard deviation.

[33] t-ratio=3.260023. Looking for a t-ratio greater than |2| is a common rule of thumb for judging significance because it approximates the 0.05 significance level.

Random Mean	Choice	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
2449	Stop	2216.73	2569.13	663.35	794.0	3639.5
2505	Straight	4849.18	3447.56	836.16	3076.6	6621.7
2543	Right	5223.5	3105.02	776.26	3568.9	6878.1
2590	Left	2919.75	2982.89	745.72	1330.3	4509.2

Table 13. Means and Standard Deviation Data for XY Fit Model

The percentile plots divide the data so that *n*% of the data are equal to or below the nth quartile. Figure 14 illustrates data shape and symmetry:

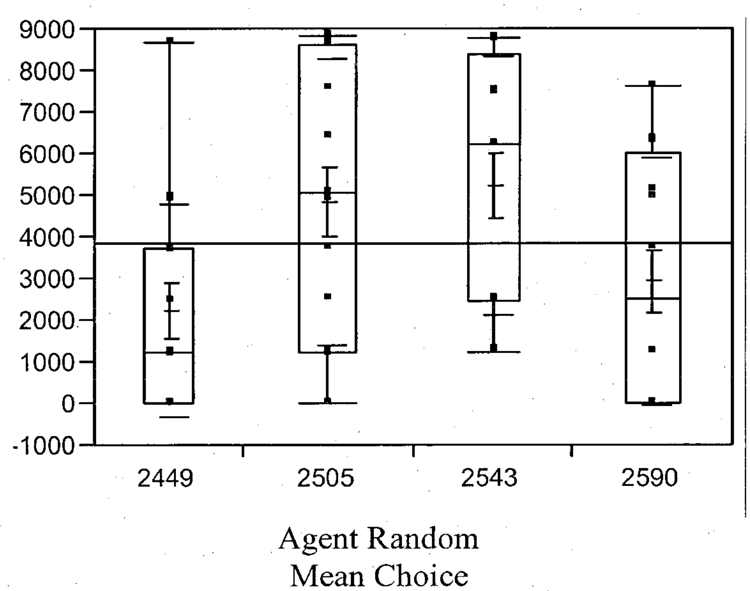


Figure 14. Fit Y by X Quartile Plot

The horizontal line inside the quartile box plot represents the median, e.g., 50th quartile. Half the values are at or below the 50th quartile and half are above. The top and bottom of the box represent the 25th and the 75th quartiles. "Whiskers" at both ends of the box extend from the end of the box to the outer-most data point that falls within 1.5

times the range from the 25th to the 75th quartile Prob > |t| 0. 0.0007. This is the probability of getting, by chance alone, a t-ratio greater (in absolute value) than the computed value, given a true hypothesis. As this value is below 0.05, this further supports a rejection of H_0.[34] Table 14shows an ANOVA sample variance of runs.

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Agent Choice	1	55824140	55824140	10.6277	0.0014
Error	126	661837357	5252677.4		
C. Total	127	717661497			

Table 14. ANOVA Sample Variance of Runs

H_0 states that in agent-object test population there are no differences between groups $Agent_{expert}$ and $Agent_{random}$. Observations reject H_0 and conclude that differences do exist. Table 15 shows power tests for type I errors.

Level / probability of Type 1 and Type II errors:

$$a = ar\ (\text{reject } H_0 \mid H_0 \text{ true})$$

$$\text{Reject } H_0 \text{ if} \mid ar_0 \mid > ar_{n-1,\, a/2}$$

$ar_{n-1,\, a/2}$ is (1- a /2) quartile of t distribution with n-1 degrees of freedom.

Power Test

a	s	d	Number	Power
0.0500	2291.872	660.3984	128	0.8989

Table 15. H_0 Power Tests for Type I Errors

[34] A value below 0.05 is interpreted as evidence that the parameter is significantly different from zero.

The sample size is 128, the effect size 660.3984 (d), and the standard deviation of the error 2291.872 (s), with a significance level of 0.05 (a). The likelihood of detecting a difference at 0.05 (a) is:

Power 0.8989 $\neq$ 0.05 (a). (The hypothesized value is the true value of the parameter)

Power 0.8989 is significantly large. (Power increases as the true parameter gets farther from the hypothesized value)[35]

Least Significant Number (LSN) tests (Table 16) define the number of observations needed to drive down the variance of the estimates enough to achieve a significant result with the given values of a, s and d.

Sensitivity Tests

a	s	d	Number(LSN)
0.0500	2291.872	660.3984	48.75644

Table 16. Sensitivity Least Significant Number (LSN) Test

This test indicates if more, such as a larger sample size, is needed to achieve significance. The results indicate:

LSN of 48.76 < n 128 (actual sample size), therefore the effect is significant.

[35] (a) = the significance level of 0.05. (s) = the standard error of the residual error in the model (Root Mean Square Error). (d) = raw effect size. Set to the square root of the sum of squares for the hypothesis divided by *n*. (*n*) = sample size - actual sample size.

There is sufficient data needed to detect the significance at 0.05 (a).

A single-degree of freedom hypothesis further tests sensitivity for how small an effect is when declared significant at *p*-value 0.05 (a). The results indicate:

Table 17 demonstrates a LSV of 801.78 = absolute value given p-value of 0.0014 (from test) = 0.05 (a). The parameter estimate 2291.88 (s) estimate > 801.78 (LSV), therefore it is significantly different from zero.

a	s	Number	LSV
0.0500	2291.872	128	801.7791

Table 17. Sensitivity Least Significant Value (LSV) Test

A single degree of freedom hypothesis further tests sensitivity for how small an effect is when declared significant at *p*-value 0.05 (a). The results indicate that there were no Type 1 or Type 2 errors observed.[36] Observations support the rejection of H_0 and conclude that differences do exist between Agent and Agent_{random}:

$E[X_i]. = 2522.625$

$\mu_0 \quad = \quad E\,[X_i].$ under $H_0 = 2522.625$

$n \quad = \quad 16$ (number of independent simulation runs)

---

[36] Type 1: Reject H_0 when it is true; P (type 1 error) = level a. Type 2: Accept H_0 when it is false; P (type 2 error) = d (1-ß), where ß = power of the test.

$X = \sum_{i=1}^{n} X_i / n$ – sample mean of runs) = 3843.42

$S^2 = \sum_{i=1}^{n} (X_i - X)^2 / (n-1)$ (sample variance of runs)

X_i = Random variable corresponding to average agent choice from the *i*th simulation run.[37]

$ar_0 = (\bar{X} - \mu_0) / (S / n^{1/2})$ is approximately *ar* (anticipated response) random variable with n-1 degrees of freedom if H_0 is true.

Level / probability of Type 1 and Type II errors:

a =ar (reject Agent and $\text{Agent}_{\text{random}}$ | H_0 true)

Reject H_0 if $| ar_0 | > ar_{n-1, a/2}$

*H_0 was not falsely rejected nor falsely accepted.

ANOVA includes a least-squares means analysis (SAS-JMP, version 6.0). The main effect (marginal) means shown are these least squares estimates. Table 18 is a display of mean variance.

	Expert Agent	Agent Random	N	DFE
Intercept	3843.42188	2522.625	64	63

Table 18. Parameter Estimates for ANOVA Fit Model

For a mean perception choice simulation value, once again averaging across all conditions, expert perceptual choices indicate more frequently that the ship would turn

37 X_i's are approximately IID normal random variables.

right $Mean_{right}$ = 5223.50 or continue straight $Mean_{straight}$ = 4849.18 (Mean of agent response = 3843.422).

Table 19 shows matched pair response values for expert agent and random agents.

Across Groups

SME Choice	Count	Mean Difference	Mean of Means
0	16	2521.2	1260.6
1	8	1287.8	1872
2	7	36.571	2492.4
3	4	-1225	3110.6
4	7	-2469	3759.8
5	7	-3759	4431.9
6	5	-4981	5035.4
7	10	-6203	5616

Test Across Groups	F Ratio	Prob>F
Mean Difference	23464.609	<.0001 within pairs Y Axis
Mean Mean	13564.836	<.0001 among pairs Y Axis

Wilcoxon Sign-Rank

	Agent Random Agent Seeded
Test Statistic	-379.00
Prob>[z]	0.010
Prob>z	0.995
Prob>z	0.005

Table 19. Matched Pair Response Values for Expert Agent and Random Agents

Observation across all test groups demonstrates the mean difference Prob>F within pairs = <.0001 and Mean Mean Prob>F within pairs = <.0001 indicating considerable unequal variances.[38] Additionally, Test Statistic Prob > |z| of 0.010 indicates that that the distributions of the two levels are not centered at the same location.[39]

Tests show that agent input variables that represent expert critical *gestalt*s, triggers and cues influence agent simulations. Accepting a method that introduces AP into HBR as a confidently accepted practice for agent simulation must show that an expert agent with AP can accurately replicate or outperform current state-of-the-art perception techniques when accomplishing critical tasks. This leads to a comparison of simulated expert, AP and state-of-the-art input and output variables.

A premise of the method to improve AP in HBR is that an agent closely mimicking a decision substantiated by a human expert should reproduce expert decisions better than current, state-of-the-art, rule-based agents. We know from psychology that experts perceive things differently than a novice [Klein *et al.* 1989]. They tend to have a perceptive "eye" for situations which the novice lacks. This is, in part, due to an expert's interpretation of the environment and effective use of cues. Implementing expert-agent decisions in their environment, and using cues from a valid meta-model, should map perception more accurately to HBR than stand alone rule-sets.

---

[38] The probability of obtaining by chance alone an F-value larger than the one calculated if, in reality, the variances are equal across all levels. Observed significance probabilities of 0.05 or less are considered evidence of unequal variances across the levels.

[39] For Prob > |z| observed significance probabilities of 0.05 or less are often considered evidence that the distributions of the two levels are not centered at the same location.

Comparisons of AP with state-of-the-art systems focus on how to build the perception agents for evaluation and how to derive the data sets. Establishing a variance between an expert system and a random interplay of the simulation objects demonstrate the influence of an experienced navigator versus that of chance. Although proven, this concept remains predictive as there is a motive to develop an expert system. A true test for the method introducing AP into HBR lies in the comparison of a state-of-the-art intelligent perception system and the AP simulation. Within this scenario, an initial inquiry establishes what action the tug/container barge would take if driven by such an intelligent system.

Current, state-of-the-art, collision-avoidance perceptions for maritime navigation reside within the sensor and software compositions of USVs [Larson *et al.* 2006]. If a tug/container barge perception agent uses an USV rule set, then a true comparison can be made to the expert AP simulation. Current, USV obstacle avoidance uses algorithms based on international rules of the road [International Navigational Rules Act 1977] and Collision Regulations (COLREGS) [U. C. G. Commandant 1972].[40]

Current rule sets used for USVs address the following events:

- Angular separation between the direction of approach of the ship contact and the Tug/container barge direction of motion ($\Delta\theta$)

[40] The USV surrogate for the tug/container barge agent incorporates COLREG Rules. Rules 7 and 8 identify collision and the action to take. Sections 4.2 and 4.3 further outline collision and avoidance rule sets. Rule 9 specifically addresses navigation in a restricted channel. Rule 14 defines avoidance and actions regarding head on collisions. Rule 15 defines the crossing collision actions. Rules 16-18 determine the hierarchy of the right-of-way. Our assumption is that the USV algorithm will always project the optimum action to avoid the approaching ship.

- Expected temporal difference between the ships and the tug/container barge in reaching the collision point (Δt)
- Distance of the ships' center to the tug/container barge at the instant of sampling (Δs) from the point of view of the tug/container barge as a surrogate USV
- Classifications of rule output actions actuate a change in the direction of tug/container barge motion (Δyd) and a change in its velocity (Δvd)
- Rules vary according to ship position areas with reference to the tub/container barge location.

USV rules applied to the experimental, AP agent simulation address the same UML interactive objects, e.g., approaching ships, according to their location-based execution of making decisions to go left, right or straight.[41] For consistency, the simulation employs the same simulation run data (Table 7) used to compare the expert system versus random change AP simulation object interplay. A new comparison creates a variance analysis using a state-of-the-art agent (USV surrogate) with the expert and random system ANOVA results. For USV representation of a container barge/tug, a meeting situation is defined if the two vessels are moving at the opposite heading within 30 degrees and approaching within 200 meters. The approaching vessel position should be on the port side of the USV container barge/tug as shown in Figure 15. This represents the final right location. For a crossing situation, the approaching vessel is moving along a heading that is between 30 and 135 degrees greater than the USV and approaches

[41] The previous decision of "stop" used as a variable for expert and agent ANOVA is compared in USV algorithms with a projected obstacle avoidance based on forward inertial distance and bearing following the tug/container barge stop action.

within 200 meters. The closest point of approach should occur on the port side of the USV (the USV should pass astern of the other vessel). This is represented in the final left location. The mutual case places the responsibility on the other vessel to avoid the USV tug/container barge. This is represented in the final straight location.

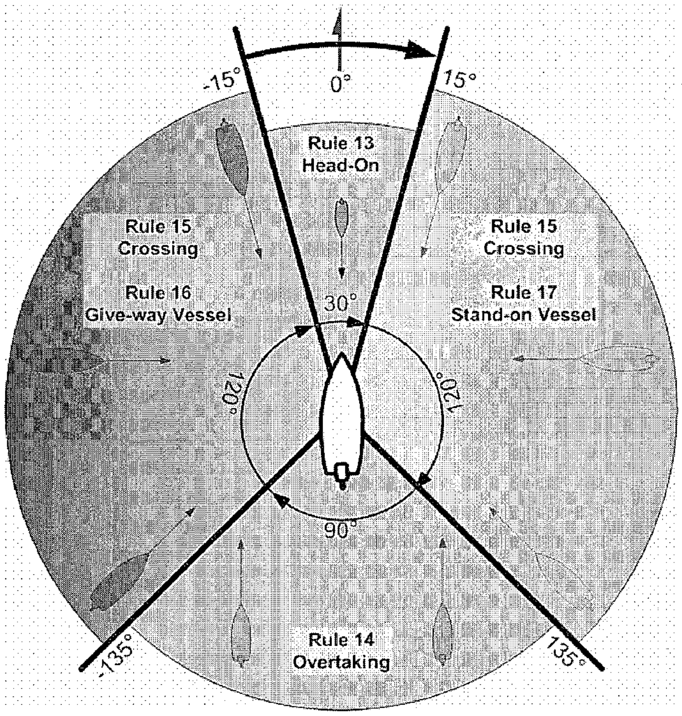


Figure 15. COLREGS Obstacle Crossing and Head-On

Scenario

In keeping with USV algorithms, anticipated tracks for obstacle avoidance of each vessel executing a left, right or straight decision, uses a projection of arched path actions inclusive of vessel advance and transfer distances. This is a visual calculation of obstacle avoidance using a reactive architecture that provides discrete to stochastic, vector to arc path conversions. USVs use these obstacle-avoidance techniques when approaching obstacle motion converts to an obstacle "vote" or "cost of avoidance" schema [Kalman 2006].

This guarantees that every pixel in an X, Y grid covers at least one arc so that all navigable paths are considered. Each of the arcs is related to the vehicle velocity and turn-rate by ($V=R/\theta$) where R is the radius of the arc, V is the vehicle velocity, and θ is the vehicle turn-rate. For the simulation, each of the decision point's path and final location use the leading edge method for identifying the leading point of obstacle avoidance [Baily *et al.* 2006]. This anticipated, approaching ship location (from the USV tug/container barge perspective) uses a radius of 200 meters, with the actual conversion located at the center of the radius, representing near field or reactive obstacle avoidance. Far field, or deliberative obstacle avoidance, was set in the simulation at an initial range of 800 meters with each vessel simulating the following model behavior:

- Speed < 8 knots
- Initial bearing 000 degrees
- Distance < 200 meters (near field)
- Distance > 800 meters (far field)
- Waypoint of 50 meters
- Approaching vessel turns of 15 degrees
- 1 Knot = 1.852 KM per hour
- Time to intercept at 600 meters = 2.42 minutes

From the point of view of the USV tug/container barge, these approaching ship objects are classified as being:

a. *Front-Right* (FR, Right θ 15^0, Left Decision)

b. *Front-Left* (FL, Left θ 15^0, Right Decision)

c. *Straight-Center* (SC, 000 ^{0}, Straight Decision)

d. *Stop* (ST, 180 ^{0}, Stop Decision).

The nature of the rules that engage objects approaching the USV tug/container barge referencing direction control, translates as: if $\Delta\theta$ is small, Δt is positive, and Δs is near, then Δyd is positive. This defines the orientation of the USV tug/container barge toward a ship that is approaching.

Calculating a collision and Priority Based Averaging (PBA) for obtaining USV tug/container barge "votes," approaching ships that have yet to be avoided are classified as obstacles and are assessed a value [Larson *et al.* 2006]. These priorities are determined as:

$$Pi = \frac{w_1 i + w_2 i}{2}$$

Here $w1i$ *is* the factor that considers the closeness of the dynamic object i from the tug/container barge, and $w2i$ considers the rate of change in the distance from the oncoming ship's center from the last period.

Avoiding collision with three possible ship positions, the tug/container barge acting as USV change in orientation is:

$$Y_d = \mathrm{s(i)}\ \Delta\varphi_{\mathrm{m}} f(\Delta s_{\mathrm{m}})$$

$$\Delta f_{\mathrm{m}} = \max |f_{i+1} - f_i|,\ i = \{1 \ldots n - 1\}$$

Where n represents the number of objects in the priority list (in this scenario, there are 3) and f_i is the angular separation of the ith object with the USV, e.g., tug/container barge heading direction. Here $s(i)$ takes a value 1 if the object i lies on the USV right and - 1 if it lies on its left. The $i + 1$th object is closer in terms of its angular separation to the USV than the ith object if it were on the USV's right and farther away from the USV than the ith object if it were on its left.

$\Delta s_m = \min \Delta s_i$ (avg), $i = 1, 2 \ldots, n$ is given where Δs_i is the normalized distance of the ith ship's center from the USV. The algorithm tries to find out the maximum free space between all adjacent ships considered from the priority list and turns along the bisector of this maximum free space. Using this formula and accessing from the leading edge of 200 meters with a normalized rate of change for the approaching ships would obtain a priority based obstacle avoidance free space vote as follows:

a. Between Front-Right and Front-Left (I - .0) (1) (.6) + (.15) = Straight vote (.85)

b. Between Front- Left and Straight-Center (I - .15) (1) (.6) + (.25) = Left vote (.7)[42]

c. Between Front-Right and Straight-Center (I - 15) (1) (.6) = Right vote (.45)

d. Between Stop and Straight-Center (I - .0) (1) (.6) = (.6).[43]

---

[42] The Straight and left turn vote options are weighted representing the far field rules of the road rule set (>800 meters) with an extremist situation at 600 yards. This represents a rule-based behavior of burdened vessels having the right of way and the turn right rule for approaching vessels. This action is standard practice for USV OA and prevents PBA scores from reflecting only free space obstacle avoidance.

[43] PBA score also take into account a probable tug/container barge forward advance of 200 meters.

Comparing rule-based results from USV obstacle-avoidance-free, space votes with expert system and random, obstacle-avoidance, simulation results allow for further observation of decision variance. If the tug/container barge perception uses an USV rule set, then a true observation of an expert system AP simulation compares the same simulation with that of an USV rule set. Higher vote counts indicate USV tug/container barge perceives the free space between "Front-Right and Front-Left" option with the greatest risk of contact. An expert or intelligent system determining action for the oncoming ships would also make a similar, obstacle-avoidance decision to alter course to right or left. This allows for a simulation and comparison of projected obstacle avoidance from the USV tug/container barge perspective using USV-obstacle, count-inverse variables. Obstacle avoidance votes symbolize projected optimum free space and anticipated approaching ship actions based on the rules of the road. Agent, rule set scores convert to the 7-point Likert scale for a like-same comparison with expert system simulation results [Dawes 2008].

Assumptions are that if the tug/container barge "acted" as a rule-based USV, what would be its actions to avoid collision? As in the method introducing AP experiment, the USV scenario objective is collision avoidance with a goal of mitigating *extremis*. As with AP, state-of-the-art rule sets, and object avoidance decisions map to the model. These variables clearly convert back to the viewpoint of a navigation expert on what action the USV would take.

Two types of agents are constructed, representing AP and rule-based parameters. AP agent behavior is anticipated to be closer to that of an expert than the rule-based agent. For both the rule-based and AP agent, 16 human expert trials are the input parameters. The correct decisions (as derived by experts) are known. This gives us:

- 16 configurations of input parameters with the desired outcomes (from the human experts)
- 16 configurations of input parameters with observed decisions of the rule-based agents (Deterministic = 16 Runs)
- 16 configurations of input parameters with observed decisions of the AP agents (Deterministic = 16 Runs)

An expected result is that the difference between the AP agent and the human expert is less than the observed difference between the rule-based agent and the human expert. In other words, how close are AP and rule-based decisions to expert decisions? We measure this by using Nearest Neighbor (NN) techniques. The idea is that a sample point is more like its neighbor than it is to the overall average. NN techniques employ Tobler's First Law of Geography - "all things are related but nearby things are more related than distant things" [Miller 2004]. In this instance, the principal remains that if an expert, or agent, takes no action, a catastrophic event will occur, e.g. collision. Using agent simulation results for AP and rule-based, turns are converted to actual NN decisions (Stop = 1, Left = 2, Straight = 3, Right = 4), as shown in Table 21.

Trial	Human Expert (Correct Choice)	HE Distance (NN)	RB Agent	RB Distance (NN)	AP Agent	AP Distance (NN)
1	1	1.00	1	5.00	1	0.00
2	2	1.73	1	5.00	2	1.00
3	2	1.73	1	5.00	2	1.00
4	2	1.73	1	5.00	2	1.00
5	3	2.83	1	5.00	2	1.00
6	3	2.83	1	5.00	3	2.00
7	3	2.83	3	4.00	3	2.00
8	3	2.83	3	4.00	3	2.00
9	3	2.83	3	4.00	3	2.00
10	3	2.83	3	4.00	3	2.00
11	3	2.83	4	2.00	3	2.00
12	3	2.83	4	2.00	4	1.00
13	4	2.00	4	2.00	4	1.00
14	4	2.00	4	2.00	4	1.00
15	4	2.00	4	2.00	4	1.00
16	4	2.00	4	2.00	4	1.00

Table 21. Nearest Neighbor Input and Output Variables[44]

The NN procedural solution is as follows:

1. Determining the parameter K = number of nearest neighbors
2. Calculating the distance between the query-instance and all the samples
3. Sorting the distance and determining the nearest neighbors based on the *K*-th minimum distance

---

[44] The human expert correct choice is an avoidance course of action. This implies that there are an additional three non-correct scores with only one correct expert response. Variation of the expert choices indicates that the well-represented cross section and validation of navigation experts in the ACTA supports the overall sampling of correct choices.

4. Gathering the category of the nearest neighbors
5. Using simple majority of the category of nearest neighbors as the prediction value of the query instance.

Table 22 is the rule set established by proximity:

Turn Value	Turn	Closest Neighbor	Turn Value	Distance (x100m)
2	If Left	Straight	3	2
4	If Right	Straight	3	2
3	If Straight	Stop	1	6
1	If Stop	Straight	1	6

Table 22. Proximity Rule Set

A prediction table provides the Euclidian distance between (a) and (b) as shown in Table 23. It is calculated and obtained directly from the distance matrix:

d (a, b) =sqrt ((xb-xa)2+(yb-ya)2))[45]

* Nearest Neigbor to HE (K=2)

Turn	X1 = Turns (HE)	X2 (distance) meter x 100	Y= Class	Turn	X1 = Turns (RB)	X2 (distance) meter x 100	RB Square Distance to query instance (X1, X2)	X1 = Turns (AP)	X2 (distance) meter x 100
1	1	6	Stop	1	6	6	$(1\text{-}6)^2 + (6\text{-}6)^2 = 25$	1	6
2	3	2	Left	2	0	2	$(3\text{-}0)^2 + (2\text{-}2)^2 = 9$	4	2
3	8	6	Straight	3	4	6	$(8\text{-}4)^2 + (6\text{-}6)^2 = 16$	6	6
4	4	2	Right	4	6	2	$(4\text{-}6)^2 + (2\text{-}2)^2 = 4$	5	2
Trials	16				16			16	

Table 23. Euclidian Nearest Neighbor Variables

[45] Where: d = distance, a = turn, b = distance SQRT(HE turn –RB turn) + HE distance – RB distance).

Results demonstrate that AP agent behavior more closely aligns with expert decisions (Figures 16 and 17). There were significant differences in the decisions of rule-based with that of human experts and AP agents. Some possible reasons for these variances include rule-based adherence to the rules of the road, COLREG burdened vessel "right of way," and "turn right" rule sets for avoidance versus human expert experience. Revealed is a general finding that strictly following navigation rules does not optimize collision avoidance.

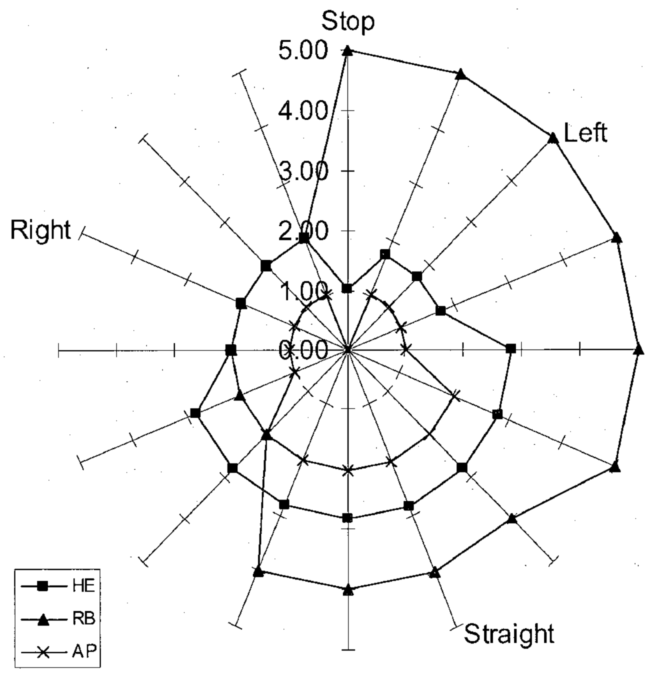
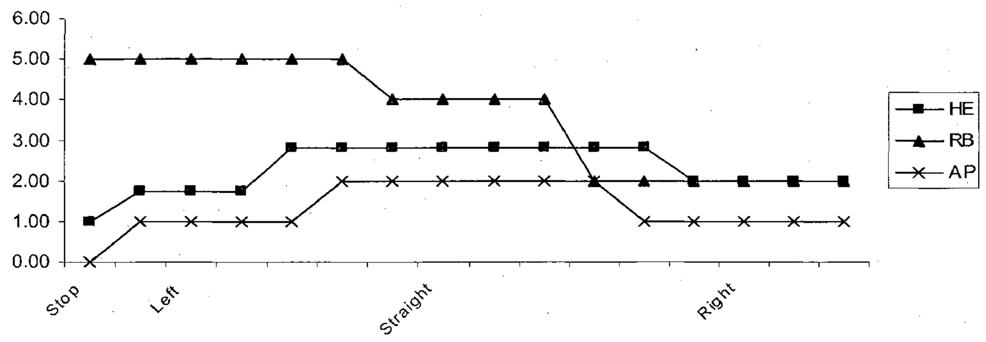


Figure 16. AP and Rule-Based Agent Alignment with Expert Decisions



Charts of: SQRT: $d(a,b)=((xb-xa)^2+(yb-ya)^2)$ by Turn

Figure 17. AP and Rule-Based Agent Alignment with

Expert Decisions

Optimum results for collision avoidance without accurate cognitive mapping of expert experience to AP are not expected. Experiments confirm that adherence to state-of-the-art rule sets may often result in different performances than those exhibited by experts. The experimental premise also supports the concept that experts have unique cognitive contributions to obstacle-avoidance behavior that are not included in rule-based systems. A significance of this work is that it provides a practical application for using proven psychological methods in games and agents. It also improves HBR in synthetic environments and allows for accurate (measurable) projections of a future state in synthetic environments. This method also allows predetermination of actions of other entities for a critical task and is a repeatable process. By using gaming techniques to express *gestalt*s for visual cues, agents have the ability to accurately "act out" projections of future states.

A comprehensive optimization analysis discovers how results are dependent upon scaling parameters, initial agent startup conditions, and variances induced by indiscriminate actions in the AnyLogic™ agent software used to conduct the agent simulation [AnyLogic™ 2006]. Sensitivity procedures involve a careful examination of how the outputs of AP as HBR varied when its inputs change [Eichelkraut and Etzkorn 2006; Gehlsen and Page 2001]. The strength of the results from the simulation runs are recorded against changes in the distribution of random variables in the system, time delay, simulation updates, optimization and parameter variation.

Sensitivity analysis ensures that agent simulation is an accurate representation of an expert system and replicates cognitive critical decisions for further comparisons with state-of-the-art rule sets. Output variables of the simulation are first demonstrated to not be random events but are associated with the cognitive input based on the critical cues given by the experts. Next, sensitivity of the simulation explores that the simulation is demonstrating the desired output based on the critical decision input variables.[46] Results of the state-of-the-art, rule-based, simulation comparisons are shown in the Appendix (Figures A-22 through A-24).

General findings are that novice ship handlers do not act like experts. Therefore, an optimum result for collision avoidance without accurate cognitive mapping of expert experience to AP is not expected. Observations confirm that state-of-the-art, rule-based performance may often be different from performance exhibited by the expert systems. This supports a concept that experts have unique cognitive contributions to obstacle avoidance behavior that are not included in rule-based systems that must be included within the framework and practical application of AP.

[46] The critical input variables include the model parameters speed, bow wave height, bow angle and distance.

An innovate use of a cognitive process for agents from knowledge solicitation and extraction of expert behavior for a critical task is to apply psychologically-accepted methods and close the gap to an applied engineering approach using this knowledge from an expert's perceived and anticipated response to a critical event for agent replication and reuse of this event structure.

This results in an increased realism of expert agent behavior as demonstrated within the AP agent domain. The use of cognitive aspects for agent, rule-set performances more closely align with expert decisions and prove feasibility, as exhibited by agent reactions to perceived cues, as compared with state-of-the-art, rule-based systems, and as demonstrated by AP agent behavior.

5. FUTURE WORK

Methods for soliciting expert cognitive knowledge and knowledge-transfer to a software engineering approach allows for reuse and adaptation to new agent domains. The method introducing AP may be tailored to new problem sets that can be accurately replicated to, and represented by, agent simulation.

Costs benefits of introducing AP are examined to improve HBR through cost avoidance associated with maritime accidents. Shipping is widely regarded as a hazardous industry as vessels sink, run aground, collide and/or catch fire. Consequently, individuals are regularly injured or killed and coastlines, livelihoods and marine life are often damaged or destroyed. Despite the fact that the frequency of maritime accidents are declining, human error continues to be a dominant factor in approximately 80% to 85% of the reported accidents [Baker and McCafferty 2005]. Failures of SA and situation assessments are the overwhelmingly predominate causal factors in the majority of accidents attributed to human error [Iarossi 2003].

Studies conducted by Lloyd's Register Educational Trust Research Unit (LRETRU) have provided measures of perceived accident risk by seafarers.[47] Results indicate that human fatigue and task omission seem closely related to failures of SA resulting in human errors and mishaps [Lloyd's 2005]. Further research reveals that collision and grounding accidents are the main risk drivers for many ship types [Bailey 2006]. Major risk reductions are achieved by measures to prevent such failure of SA as it

---

[47] LRETRU used available maritime datasets from 2000 through 2005.

applies to the safety of ship navigation. If the HBR AP model were useful in reducing risks associated with critical navigation incidents through increased SA, then the cost benefit would result in a reduction of losses associated with the diminution of human error.

It is possible to assess the scope of the risk by viewing the number, type and nature of incidents and the associated costs. Tables 20 through 22 are statistics accumulated by Lloyds for incidents, number of ships, and percentage of incidents recorded between the years 2000 and 2005 as recorded by the seven registers.[48]

Type of Incident	**Combined Number**
Collision	4,723
Contact	1,000
Grounding	1,834
Fire	435
Explosion	22
Fire/Explosion	269
Sinking	622
Total 8,905	Total 8,905

Table 20. Total Incidents Recorded by the Seven Registers between 2000 and 2005

[48] Seven registers refers to the seven largest maritime registries required by the Merchant Shipping Act of 1976.

Year	Total Number of Ships
2000	6,985
2001	7,212
2002	14,859
2003	14,566
2004	14,542
2005	7,777
Total	65,941

Table 21. Total Number of Ships on the Seven Registers

Type of Incident	Percentage of incidents	Rate per 1000 ships
Collision	7.16	71.6
Grounding	2.78	2.78
Contact	1.52	1.52
Sinking	0.94 [LR sinking's = 0.075].[49]	9.4 [LR = 0.75].
Fire	0.82	8.2
Explosion	0.04	0.4

Table 22. Percentage of Incidents over the Period 2000-2005

Based on available data, during the period 2000 through 2005, the type of incident that a seafarer is most likely to experience (of those listed) is ship collision This is a prospect that 7.16% of all ships listed in the seven registers will experience an incident over the five-year period and will be involved in ship collision.[50] Note that ship collision is the same incident identified by SMEs as a critical task during the ACTA. This episode

[49] As cited in Lloyd's *Register World Casualty Statistics* for 2005.

[50] Ship registries are maintained by approximately 30 nations and are open to ship-owners from all nations for choosing where to "flag" their ships as a mandate of international law. The seven top registries reflect approximately 53 % of the ships are registered.

provides a solid foundation for the critical ship scenario simulated in the experiment. Regarding the scope of this problem domain, at the end of 2005, the international trading fleet consisted of approximately 46,222 reported ships in the registries, combining 597,709,000 gross tones [Lloyd's 2005]. The bulk of the fleet consists of (Table 23):

Ship Type	Total Number of Ships
Cargo ships	18,150
Tankers	11,356
Bulk carriers	6,139
Passenger ships	5,679
Container ships	3,165
Other ship types	1,733
Total	46,222

Table 23. Registered International Trading Fleet Ships 2005

Specific locations of a ship at the time of an accident are categorically listed by Lloyds to include "port" for accidents that occurred within the area of a port, "overseas" for accidents that occur at sea and at a considerable distant from the coast, and "controlled seaways" for straits and canals [Lloyds 2005]. Percentages associated with the location of reported incidents include 47.4% for ports, 33.5% for overseas and 19.1% for controlled seaways. Baker notes that the most frequent accidents in "ports" are grounding and collisions [Baker and McCafferty 2005]. With the highest percentage of reported shipping accidents occurring in "ports," the same environment used in the HBR AP approach, further provides verification for the "use case" as affirmed by SMEs in the ACTA.

How can improved SA, by introducing AP to HBR, contribute to reduced accident costs? The answer can be found by viewing the effects of introducing improved ship navigation SA technology and observing the related reductions in shipping accident-incident rates. For example, coupling electronic navigation charts with smart risk detection information systems has proven effective for reducing costs by approximately 38% for those incidents associated with grounding and collision [Vanem *et al.* 2007]. One example of a computational navigation chart is the Electronic Chart Display and Information System (ECDIS). This system is an evolved form of computer-based navigation information system that is also in compliance with International Maritime Organization (IMO) regulations for shipboard navigation [IMO 2007]. This system has expanded the Electronic Chart System (ECS) to include risk avoidance vectoring and collision alert notification based on relation to a vessel's position and movement. Integration of HBR AP interface for instant recall of critical scenario data validated by expert ship handlers into the Electronic ECDIS would provide a significant risk reduction and support a follow-up savings through cost avoidance. This is a significant possibility that spans the entire shipping fleet, as ECDIS and AIS (Automatic Identification Systems) are required aboard all flagged, ocean-bound merchant vessels [SOLAS 1974].

How would this work? ECDIS systems integrate Electronic Navigational Charts (ENC) and Global Positioning System (GPS) data and other sensor data, (e.g., sonar, radar, AIS, fathometer, etc.) into a Common Operating Picture (COP). They also integrate and display navigational operational and information data. Predetermined, critical, scenario results established using the AP method could be added to the ECDIS informational database. Critical scenario results could be cued automatically for reference, either when a navigator is alerted of a dangerous shipping environment or when a navigator has determined through experience and judgment that a precarious real-time scenario exists. In either case, these critical-scenario vector charts would be available in a database, formatted from EDCIS geo-reference objects, and integrated into the COP.

If increasing navigational SA through the introduction of ENC and smart, risk-detection systems results in a reduction of ship accident costs by approximately 38%, then it infers that a further increase in SA will result in an increased reduction of these costs. What are the ship accident costs? One method to ascertain ship accident costs is to view insurance loss statistics associated with maritime operations. Table 24 is an example provided by the International Union of Marine Insurance in 2006 for the total loss of a 10,000 Transport Equivalent Unit (TEU) container vessel: [Kirchner 2006].

Value of the vessel	EUR 200,000,000
Costs	EUR 2,500,000
Average value per 20 ft container:	EUR 20,000
Average value of cargo	EUR 200,000,000
Containers (Hull):	EUR 20,000,000
Total Avg. Claim	EUR 422,500,000 [51]

Table 24. Total Loss of a 10,000 TEU Container Vessel

Lloyd reported in 2006 that its gross premiums for insurances paid were $2,585,146,593 USD. (Note that this does not represent costs related to the total world shipping fleet, however, it does represent a cross section from a large, world, maritime

[51] Note: At the time of this printing, one Euro equates to 1.45 USD.

insurance corporation.) There were paid claims for $561,504,032 USD with at total outstanding claim of losses totaling $1,650,240,070 USD for a total of $2,211,744,102 USD [Lloyd 2005]. If 7.16 % of the claims relate to collision, then a related cost estimate for claims in 2006 is approximately $158,360,877.70 USD. The same reduction of ship accident costs by 38% is not expected and is noticed previously through the introduction of smart navigations' systems that would be realized by integrating the HBR AP model. However, if this integration into ECDIS resulted in a minimal 1% increase of SA, it would have realized in a cost avoidance of approximately $1,583,608.78 USD for 2006.

Problems with SA are not unique to maritime accidents. A review of military aviation mishaps and the concurrent study of accidents among major air carriers discovered the absence of effective SA as a leading causal factor. In this study, 88% of those involving human error are attributed to problems associated with SA [McNeese and Vidulich 2002]. Developing computational cognitive models as spatial, temporal or other reasoning agent instruments could provide cost benefits to the aviation community analogous to those discussed for the maritime community. Similarly, an integration of the HBR AP model into air navigation systems, e.g., autopilot or collision warning system, would alert the aviator of "expert defined" dangerous environments or could provide a reference from the HBR AP model when the pilot has determined, through experience and judgment, that a threatening environment exists.

Future possibilities should consider the practicality of integrating the AP process into medium and large simulations [Bosse and Jonker 2006]. Generally, larger simulations employ a storyboard that provides background information for the scenario.[52] These are generally comprised of storylines defined by events that guide and shape the scenario. In a typical simulated event, there are multiple storylines operating at once, sequentially and parallel, culminating in events to achieve an experiment objective [Joines *et al.* 2000]. Figure 18 is an example of standard simulation components.

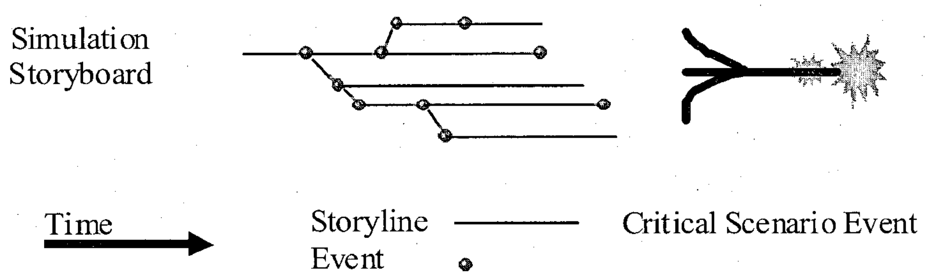


Figure 18. Typical Simulation Storyboard Components

Characteristically, multiple storylines are developed with each defined event. Together, they support the overall simulation's goal directed behavior [Chandrasekaran and Josephson 1999]. It is plausible that certain storylines and events have critical tasks associated with the simulation goal. Future work could involve identifying these simulation critical tasks and injecting AP methods using expert working memory to influence the model behavior (Figure 19). Accurate measurement of anticipated events could augment the model's V&V and simulation credibility.

[11] In this instance, "medium to large" is considered a simulation with no less than 1,000 participants.

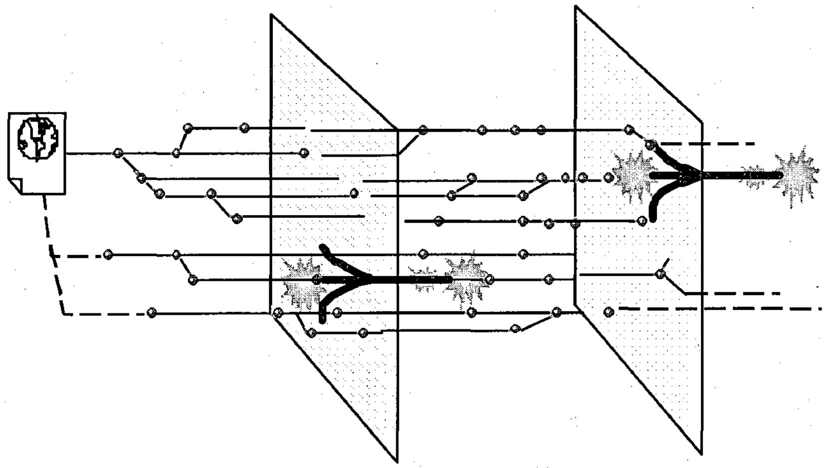


Figure 19. Injecting Critical Events into Simulation Storyboard

Once experts have identified critical tasks associated with the storylines, simulation test scores from SME estimates of anticipated actions obtained from observing a replica-critical scenario can be recorded. This data could then be analyzed using methods similar to the AP approach (expert agent) by perception and then injected into the simulation.

Another future endeavor would be to observe simulated time "state space" in a synchronous mode using perception [Zeigler *et al.* 2000]. For example, the study of quantum structures note that a state space is ever changing and contains many solutions and these differing solutions become apparent, based on the instance of the observation [Schrödinger 1995]. Most simulation times are set at regular intervals and do not consider decision-making observations as "space-time" variables that might actually alter the outcome of a state space based on the timing of the observation itself, possibly rendering results with different outcomes. Because of this phenomenon, it remains difficult to validate simulations for objects represented on the nanoscale[53] [Stormer *et al.*1999]. In the world of nanoscience, each observation is different and actually becomes dependent on the viewpoint of the observer. There is little reference found for experimentation through altering simulation time steps and measuring cognitive perception variances. Expanding the AP method to experiment with altering model time steps and observing these variances would prove an interesting endeavor.

Concerning scenario-driven simulations, empirical regularities occur during experimentation because static treatment of software input data ultimately results in a stationary distribution [Kelton 2000]. In addition, effective, empirically-based models depend on high quality datasets. These datasets are consistently preselected and are constructed according to criteria that reflect certain choices. Consequently, these choices are often biased. Future work using AP methods could help reduce bias and provide a value added by injecting "real time" observational data based on expert knowledge derived from long term memory. Results could provide realism to a dynamic simulation process using this expert derived object data.

[53] A nanoscale is smaller than a micron, on the scale of large molecules (one millionth of a meter).

6. CONCLUSION

Application of psychologically accepted methods outlined in this thesis result in a new technique of knowledge solicitation enabling an improved way to develop and document rule sets for agents. Within these methods, the cognitive aspects are particularly important. The methods applied, from the identification of critical tasks to the V&V based on human expert data sets, contribute to the greater body of knowledge.

Findings may be replicated in other domains by using this approach, specifically the methods provided in Chapter 3. Reuse of these methods requires that the domain selected involves a critical task or hazardous environment where the task entails expert decision-making such as those to avoid accident, injury or destruction. Next, domain experts must be identified and willing to participate. A briefing for experts is recommended on the entirety of the experiment as well as disclosure of all information collected. This includes results provided regarding their participation in the cognitive testing. ACTA may be used to identify a task that experts agree require skills significantly beyond what a novice could provide to safely perform the task. ACTA also provides a way to validate that experts are, in fact, SMEs.

Using knowledge extracted from ACTA and SAGAT, SMEs may develop a scenario representing the single critical task with a particular focus on perception of critical events. Cognitive maps provide a good outline for the sequencing of scenario events and the optimum choices an expert must make based on critical cues and resulting triggers. SMEs must also agree on both the exact timing and the specific event structure that a critical choice has to be made in order to safely avoid endangerment. This becomes the decision point where an expert must anticipate his or her future action and is the "freeze frame" used for SART. Gaming techniques prove to be a good collaboration platform for both SMEs and the developer in building: a replica of the scenario, identifying important cognitive elements associated with the task transferring specific objects identified for turning the scenario into a critical event, and more specifically, in observing the interaction and building visual representations for *gestalt* critical cues and event time lines for triggers associated with the significant events.

A scenario replicating this task is now able to function as both a gaming vignette to test SME-anticipated actions and as a template to build an engineering model for an agent domain. A meta-model is recommended for the creation of the scenario in order to categorize object relationships and to help apply logic and reasoning to scenario objects needed for a UML approach. A goal is to use this information to construct an agent domain that replicates the critical task environment and mimics the behavior of the expert. Suggestions include using the same participating SMEs to validate that the agent scenario is, in fact, a true representation of the critical event. SART cognitive scores can also contribute to building the expert agent rule-sets.

These methods help to close the gap between an expert's knowledge of, and perceived response to, a critical event which is extracted by using accepted psychological methods, to replicate and reuse this event structure by means of an applied engineering approach. Generally, this accomplishes:

- Bridging the knowledge of an individual by including a human science process
- Demonstrating effective use of a human science tool for extracting this knowledge
- Achieving favorable results by using an AP domain example of critical task.

Supporting this proof of concept, findings demonstrate that the AP agent domain is a good domain for knowledge transfer. Psychological models support cognitive concepts of perception and anticipation. Many of these models have established knowledge extractions procedures. These procedures allow for identifying underlying logic and reason used in critical-task, decision-making process. Observation of a critical navigation task further supports a feasibility of using this extracted expert knowledge for a specific critical "use case" able to be both validated by experts and used to develop the agent architecture.

Expert knowledge obtained from cognitive perception test results is transferable to an agent domain that represents an expert knowledge structure. Substantiation for closing this psychology to engineering gap is the observation that an agent in the AP demonstration domain who is using cognitive aspects for perception that are obtained from psychological tests observed behavior better represents the actual expert decisions than so state-of-the-art, rule-based, perception agents.

BIBLIOGRAPHY

Anderson, J. R. and C. Lebiere (1998), *The Atomic Components of Thought*, Erlbaum, Mahwah, NJ.

Anderson, J. A. and E. Rosenfeld, Eds. (1988), *Neurocomputing*, MIT Press, Cambridge, MA.

Anderson B., Peissig J. J., Singer J. and D. L. Sheinberg (2006), "XOR Style Tasks for Testing Visual Object Processing in Monkeys, *Vision Research*, 46, pp. 1804-1815.

Annett, J. (2000), "Theoretical and Pragmatic Influences on Task Analysis Methods," Schraagen, J.M., Chipman S.F., and V. L. Shalin, Eds., *Cognitive Task Analysis,* Lawrence Erlbaum Associates, Mahwah, NJ, pp. 25-37.

AnyLogic™ Version 5.3.1 (2006), Anylogic Agent Software, Montigny sur Loing, France.

Aristotle (350 B.C.E.), "Human Knowledge," In *On the Soul*, translation by J.A. Smith, Book III, Internet Classics Archives, Cambridge, MA.

Bacchus, F. and J. Grove, J. Halpern and D. Koller, (1993), "Statistical Foundations for Default Reasoning." In *Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI)*, Chambery, France, August 1993, pp. 563-569.

Bailey, N. (2006), "Risk perception and Safety Management Systems in the Global Maritime Industry," In *Policy and Practice in Health and Safety*, 4 (2) pp. 59-75.

Bailey, T. and E. M. Nebot, J. K. Rosenblatt and H. F. Durrant (2006), "Behaviour-Based Docking using the DAMN Arbiter," Department of Mechanical and Mechatronic Engineering, The University of Sydney, NSW.

Baker, C. and D. McCafferty (2005), "Accident Database Review of Human-Element Concerns; What do the Results mean for Classification?" As presented to the Royal Institution of Naval Architects, In *Proceedings of the Human Factors in Ship Design, Safety and Operation Conference*, London, U.K., February 23-24.

Balci, O. (2004), "Quality Assessment, Verification, and Validation of Modeling and Simulation Applications," In *Proceedings of the 2004 Winter Simulation Conference*, Washington, DC, IEEE, Piscataway, NJ, pp. 122-129.

Balci, O. (2003), "Verification, Validation, and Certification of Modeling and Simulation Applications," In Proceedings *of the 2003 Winter Simulation Conference*, New Orleans, LA., IEEE, Piscataway, NJ, pp. 150-158.

Balci, O., Richard E. Nance, James D. Arthur, and William F. Ormsby (2002), "Expanding Our Horizons in VV&A Research and Practice," In *Proceedings of the 2002 Winter Simulation Conference*, San Diego, CA., IEEE, Piscataway, NJ, pp. 653-663.

Balci, O. (1997), "Verification, Validation and Accreditation of Simulation Models," In *Proceedings of the 29th Winter Simulation Conference*, Atlanta, GA, 7,10, pp. 135 -141.

Balci, O. and R.G. Sargent (1984), "Validation of Simulation Models via Simultaneous Confidence Intervals," In *American Journal of Mathematical and Management Science*, 4.3, pp. 121-30.

Bandini, S., S. Manzoni and C. Simone (2002), "Mechanisms to Support Situated Agent Systems," In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS, part 3, July, pp. 1183-1190.

Basadur, M.S., J. Conklin, and G.K. VanPatter (2007), "Rethinking Wicked Problems, (Part 1)," In *Next Design Leadership Institute Journal,* 10.1.

Beer, R. D. (2003), "The Dynamics of Active Categorical Perception in an Evolved Model Agent," *Adaptive Behavior*, 11, 4, pp.209-243.

Bergmann, M., J.H. Moor and J. Nelson (2003), *The logic book (4th ed.),* McGraw-Hill, Columbus, OH.

Bonecito C. and K. Burns (2003), "Mapping the Mountains; A Survey of Cognitive Engineering Methods and Uses," In *Proceedings of the 6th Conference on Naturalistic Decision Making*, Pensacola, FL.

Booch, G., (2003), *Object-Oriented Analysis and Design with Applications* (3rd Edition), Addison Wesley Longman Publishing Co., Boston, MA.

Boole G. (1854), *An Investigation of the Laws of Thought on Which are Founded the Mathematical Theories of Logic and Probability* (1854), Walton and Mayberly, Cambridge-Macmillan, London, UK.

Borrie, D., S. Isnandar and C.S. Ozveren (2006), "Simulation of Complex Environments: The Fuzzy Cognitive Agent," In *Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications*, Vancouver, Canada, pp. 348-353.

Bosse T., and C. M. Jonker (2006), "An Integrative Modeling Approach for Simulation and Analysis of Adaptive Agents," In *Proceedings of the 39th Annual Simulation Symposium,* Huntsville, AL., pp. 312-319.

Botelho, S., S. Willmott, T. Zhang, and J. Dale (2002), "Review of Content Languages Suitable for Agent-Agent Communications", Technical Report.

Braine, M.D.S. (1978), “On the Relation Between the Natural Logic of Reasoning and Standard Logic,” In *Psychological Review*, 85, pp. 1-21.

Breese, J. S. (1992),"Construction of Belief and Decision Networks," In *Computational Intelligence*, 8, 4, pp. 624–647.

Butz, M., O. Sigaud, G.Pezzulo and G. Baldassarre (2007), “Anticipations, Brains, Individual and Social Behavior: An Introduction to Anticipatory Systems,” In *Anticipatory Behavior in Adaptive Learning Systems,* Vol. 4520, pp. 1-18.

Brutzman, D., Morse, M. J., Pullen, M. and M. Zyda (2002), “Extensible Modeling and Simulation Framework (XMSF): Challenges for Web-Based Modeling and Simulation,” In *Interim Technical Report, Findings and Recommendations Report: Technical Challenges Workshop, Strategic Opportunities Symposium*, Naval Postgraduate School, Monterey, CA.

Bureau of Transportation Services (2005), U.S. Department of Transportation, Research and Innovative Technology Administration*, Journal of Transportation and Statistics*, ISBN 1094-8848.

Cacioppo, J.T., Ed. (2002), *Foundations in Social Neuroscience*, MIT Press, Cambridge, MA.

Camerer, C. (2003*), "Behavioral Game Theory: Experiments in Strategic Interaction,*" In *Roundtable Series in Behavioral Economics*, Princeton University Press, NJ.

Carson, J.S. (1993), “Modeling and Simulation World Views,” In *Proceedings of the 1993 Winter Simulation Conference*, pp. 18-23, Los Angeles, CA.

Cassimatis, N., Gregory, J., Bugajskab, M., and A. Schultzb (2004), “Integrating Cognition, Perception and Action through Mental Simulation in Robots,” In *Robotics and Autonomous Systems,* 49, pp. 13-23.

Castelli, F., F. Happe, U. Frith, and C. Frith (2000), “Movement and Mind: A Functional Imaging Study of Perception and Interpretation of Complex Intentional Movement Patterns,” In *Neuroimage,* 12, 3 pp. 314-25.

Chakravarthy, A. and D. Ghose (1998), “Obstacle Avoidance in a Dynamic Environment: A Collision Cone Approach,” In *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, Vol. 28.

Chandrasekaran, B. and J.R. Josephson (1999), “Cognitive Modeling For Simulation Goals: A Research Strategy for Computer-Generated Forces,” In *8th Computer Generated Forces Conference*, 8TH-CGF-100.

Chaturvedi, A. R., D.Snyder, C.M. Foong and B. Armstrong (2005), "Bridging Kinetic and Non-Kinetic Interactions Over Time and Space Continua," In *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference* (I/ITSEC), Paper 2123, pp. 1-11, Orlando, FL.

Chen, G. and B.K. Szymanski (2001), "Component Oriented Simulation Architecture: Toward Interoperability and Interchangeability," In *Proceedings of the 2001 Winter Simulation Conference*, Arlington, VA. , 1, pp. 495-501.

Cheng, P. W. (1997), "From Co variation to Causation: A Causal Power Theory," In *Psychological Review*, 104, pp. 367-405.

Chomsky, N. (1957), *Syntactic Structures*, Mouton, The Hague.

Chery, S. and P.S. Farrell (1998), "A Look at Behaviorism and Perceptual Control Theory in Interface Design," In *Defense and Civil Institute of Environmental Medicine*, 98, R-12.

Chick, S., P. Sanchez, D. Ferrin, and D. Morrice (2003), "Models For Continuous and Hybrid System Simulation," *In Proceedings of the 2003 Winter Simulation Conference*, 2, 03CH7499.

Cicirelli F., A. Furfaro, A. Veit Elser, and L. Nigro (2007), "An Agent Infrastructure for Distributed Simulations over HLA and a Case Study Using Unmanned Aerial Vehicles," In *Proceedings of the 40th Annual Simulation Symposium*, Norfolk, VA., pp. 231-238.

Corker, K. M. and B.R. Smith (1993), An Architecture and Model for Cognitive Engineering Simulation Analysis: Application to Advanced Aviation Automation, In *Proceedings of the AIAA Computing in Aerospace Conference* 9, San Diego, CA.

Croft, D., and P. Thagard (2002), Dynamic imagery: A Computational Model of Motion and Visual Analogy, In *Model-based Reasoning: Scientific Discovery, Technological Innovation Values,* Kluwer-Plenum, New York, NY, pp. 259-274.

Craik, K. (1943), The *Nature of Exploration*, Cambridge University Press, Cambridge, UK.

Dale, A. M. and E. Halgren (2001), "Spatiotemporal Mapping of Brain Activity by Integration of Multiple Imaging Modalities," *Current Opinion in Neurobiology*, 11, pp. 202-208.

Davis, K. P. and A. R. Anderson (2003), *Improving the Composability of Department of Defense Models and Simulations*, RAND, Santa Monica, CA.

Dawes, J. (2008), "An Experiment using 5 Point, 7 Point and 10 Point Scales," *International Journal of Market Research*, 50.1.

Dayton, T. (1993), "Skills Needed By User-Centered Design Practitioners in Real Software Development Environments: Report on the CHI `92 Workshop." In *SIGCHI Bulletin*, 25, 3, pp.16-31.

Dehaene, S., E., Pinel, P., Stanescu, R., and S. Tsivkin (1999), “Sources of Mathematical Thinking: Behavioral and Brain-imaging Evidence,” In *Science*, 284, pp. 970-974.

DoD Directive 5000.59 (2007), Defense Modeling and Simulation (M&S) Management, U.S. Department of Defense.

Defense Modeling and Simulation Office DMSO (2003), “Special Topic - Validation of Human Behavior Representations, Verification, Validation, and Accreditation (VV&A),” In *Recommended Practices Guide (RPG),* U.S. Department of Defense.

Dennis, M. J., and W. Ahn (2001), "Primacy in Causal Strength Judgments." In *Memory & Cognition,* 29, pp.152-164.

Deutsch, S.E. And R.W. Pew (2002), “Modeling Human Error in a Real-World Teamwork Environment,” In *Proceedings of the Twentieth-fourth Annual Meeting of the Cognitive Science*, Fairfax, VA., pp. 274-279.

Druin, A. and C. Solomon (1996), *Designing Multimedia Environments for Children,* Wiley, NewYork, NY.

Edelman, G.M. (1989), The *Remembered Present: A Biological Theory of Consciousness*, Basic Books, New York, NY.

Eichelkraut, C., and L. Etzkorn (2006), “Optimization of Military Distributed Control Systems using a Mobile Agent-based Approach,” In *Proceedings from the Agent Directed Simulation 2006 Conference*, Huntsville, AL.

Einstein, A. (1936), "Physik und Realität," *Journal of The Franklin Institute*, 221, pp. 313-347.

Ekdahl, B. (1994), "The concept of models in anticipatory agents," In *Technical Report LU–CS–TR: 94,* Dept. of Computer Science, Lund University, Lund, Sweden.

Ellis, W.D. (1938), *A Source Book of Gestalt Psychology*, Harcourt, Brace and World, New York, NY.

Endsley, M.R., Bolté, B. and D. Jones (2003), *Designing for Situation Awareness*, Taylor and Francis, New York, NY.

Endsley, M.R. (1996), “Situation Awareness Measurement in Test and Evaluation,” O'Brien, T.G, and S. G. Charlton, Eds., In *Handbook of Human Factors Testing and Evaluation,* pp.159-180, Lawrence Erlbaum, Mahwah, NJ.

Endsley, M.R. (1995), "Toward a Theory of Situation Awareness in Dynamic Systems," In *Human Factors*, 37(1), pp. 32-64.

Engels, G., R. Heckel and J. Küster (2001), "Rule-Based Specification of Behavioral Consistency Based on the UML Meta-model," In *The Unified Modeling Language; Modeling Languages, Concepts, and Tools*, Springer, Heidelberg, Germany pp. 272-286.

Felleman, D., and D. Van Essen (1991),"Distributed Hierarchical Processing in the Primate Cerebral Cortex." Cereb Cortex 1:1-47.

Fink, P., P. Foo and W. Warren (2007), "Obstacle Avoidance During Walking in Real and Virtual Environments," *ACM Transactions on Applied Perception* (TAP), 4, 1, ACM Press, New York, NY.

Finke, R. (1989), Principles *of Mental Imagery*, MIT Press, Cambridge, MA.

Firby, J. (1987), "An Investigation into Reactive Planning in Complex Domains," In *Proceedings of the National Conference on Artificial Intelligence (AAAI),* pp. 202–207.

Fisher, R.A. (1925), Statistical *Methods for Research Workers*, Oliver and Boyd, Edinburgh, UK.

Fishwick P. (1994), Simulation *Model Design and Execution: Building Digital Worlds*, Prentice-Hall, Inc, Englewood. Cliffs, NJ.

Fishwick, P.A. and J.A. Miller (2004), "Ontologies for Modeling and Simulation: Issues and Approaches," In Proceedings *of the 2004 Winter Simulation Conference*, 1, pp. 250-264.

Freeman, J. T. and M.S., Cohen (1998), "A Critical Decision Analysis of Aspects of Naval Anti-Air Warfare," In *Cognitive Technologies, Inc.*, Feb. Technical Report.

Garrett, R.B., (2007), "Effects-Based Approach to Operations (EBAO) Experimentation for Validating Human Behavior Representation (HBR) in Simulated Environments," In *Spring 2007 Simulation Interoperability Workshop*, Norfolk, VA., 07S-SIW-050.

Garrett, R.B, (2006), "Validation of Human Behavior Representation (HBR) Agent-based Simulation of Anticipatory Systems," In *Proceedings of the Winter 2006 Simulation Interoperability Workshop,* ADS 2006, Orlando, FL.

Gaskins III, R.C., C.M. Boone, T.M. Verna, and M.D. Petty (2004) "Cognitive and Behavioral Psychological Research for Crowd Modeling, "In *Proceedings of the NATO Modeling and Simulation Group Conference*, Koblenz, Germany.

Gavrila, D.M. and L.S. Davis (1995), "Towards 3D Model-Based Tracking and Recognition of Human Movement: A Multi-View Approach," In *Proceedings of the Int'l Workshop Automatic Face- and Gesture-Recognition*, Zurich, Switzerland.

Gehlsen, B. and B. Page, (2001), "A Framework for Distributed Simulation Optimization," In *Proceedings of the 33nd Conference on Winter Simulation*, Arlington, Virginia , pp. 508-514.

Geoffrion, A. (1987), "An Introduction to Structured Modeling," *In Journal of Management Science*, May, pp. 547-588.

Ghose, A.and G. Koliadis (2007), "Actor Eco-systems: From High-Level Agent Models to Executable Processes via Semantic Annotations," In *Proceedings of the 31st Annual International Computer Software and Applications Conference* (COMPSAC 2007), Beijing, China, 2, pp. 177-184.

Gibson, J. (1966), *The Senses Considered as Perceptual Systems*, Houghton Mifflin, Boston, MA.

Gibson, J. (1979), *The Ecological Approach to Visual Perception*, Houghton Mifflin, Boston, MA.

Gigerenzer, G., U. Hoffrage and H. Kleinbölting (1991), "Probabilistic Mental Models: A Brunswikian Theory of Confidence," In *Psychological Review*, 98, pp.506-528.

Gigerenzer, G., and D. J. Murray (1987), *Cognition as Intuitive Statistics*, Erlbaum, Hillsdale, NJ.

Gilovich, T., D. Griffin and D. Kahneman (2002), *Heuristics and Biases: The Psychology of Intuitive Judgment*, Cambridge University Press Cambridge, Cambridge, MA.

Giordano, J.C., P.F. Reynolds and D.C. Brogan (2004), "Exploring the Constraints of Human Behavior Representation," In *Proceedings of the 36th Winter Simulation Conference: Military Applications Session, Human Systems Modeling*, Washington, D.C., pp. 912-920.

Glasgow, J. I., and D. Papadias (1992),"Computational Imagery," In *Cognitive Science*, 16, pp. 355-394.

Glover, F., M. Laguna, and R. Marti (2003), "Scatter Search, Advances in Evolutionary Computing," In *Theory and Applications*, Springer-Verlag, New York, NY, pp. 519-537.

Goldstein, E. (2002), *Sensation and Perception*, Sixth Edition, Wadsworth, Pacific Grove, CA.

Goldstein, G., and M. Hersen (1984), *Handbook of Psychological Assessment*, Pergamon Press, New York, NY.

Genesereth, M.R. and R. E. Fikes (1991), “Knowledge Interchange Format, Version 3.0 Reference Manual”, In *Technical Report Logic-92-1*, Stanford, CA.

Goldvarg, E. and P. N. Johnson-Laird (2001), “Naive Causality: A Mental Model Theory of Causal Reasoning and Meaning,” In *Cognitive Science*, 25, pp. 565-610.

Grill-Spector, K., Z. Kourtzia and N. Kanwisher (2001), “The Lateral Occipital Complex and its Role in Object Recognition,” In *Vision Research*, 41, pp.1409-1422.

Groenewegen, H. J. and H. B. M. Uylings (2000), “The Prefrontal Cortex and the Integration of Sensory, Limbic and Autonomic Information,” In *Cognition, Emotion, and Autonomic Responses*, pp. 3-28.

Gross, D., D. Pace, S. Harmon and W. Tucker (1999), “Why Fidelity?” In *Spring 1999 Simulation Interoperability Workshop*, 99S-SIW-168.

Graham, I. (1991), *Object Oriented Methods*, Addison-Wesley, Wokingham, UK.

Grassi, C. R. (2000), “A Task Analysis of Peir Side Ship-Handling for Virtual Environment Ship-Handling Simulator Scenario Development,” M.S. Thesis, Naval Postgraduate School. Monterey, CA.

Harmon, S. Y. (2002), “A Taxonomy of Human Behavior Representation Requirements,” In *Proceedings of the 11th Conference on Computer Generated Forces and Behavior Representation*, 02-CGF-024.

Harvey, I., E.A. Paolo, E. Tuci, R. Wood and M. Quinn (2005), “Evolutionary Robotics: A New Scientific Tool for Studying Cognition,” *Artificial Life*, 11, pp. 79-98.

Hawkins, J. and S. Blakeslee (2004), *On Intelligence: How a New Understanding of the Brain will Lead to the Creation of Truly Intelligent Machines*, Times Books-Henry Holt and Company, New York, NY.

Hecker, C. (1996), “Physics, The Next Frontier,” In *Game Developer*, October-November, pp. 12-18.

Hecht-Nielsen, R., McKenna, T. (2003), *Computational Models for Neuroscience*, Springer-Verlag, London, UK.

Heijden, V., R. Duin, D. Ridder, and D. Tax (2004*), Classification, Parameter Estimation and State Estimation*: *An Engineering Approach Using MATLAB,* First Edition, John Wiley and Sons, Hoboken, NJ.

Herrero, P., A. Antonio, S. Benford, and S. Greenhalgh (2002), "Increasing the Coherence between Human Beings and Virtual Agents," In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems*, Bologna, Italy, pp. 354 – 355.

Hill, R.W., Y. Kim and J. Gratch (2002), "Anticipating Where to Look: Predicting the Movements of Mobile Agents in Complex Terrain," In *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-agent Systems*, Bologna, Italy, 2.

Hoffman, R., B. Crandall, and N. Shadbolt, (1998), "Use of the Critical Decision Method to Elicit Expert Knowledge: A Case Study in the Methodology of Cognitive Task Analysis," In *Human Factors,* 40, 2, pp. 254-276.

Hogeweg, P. (1989), "Mirror Beyond Mirror, Puddles of Life," In *Artificial Life, C. Langton, Ed,* pp. *297*-316.

Holyoak, K. J. and D. Simon (1999), "Bidirectional Reasoning in Decision Making by Constraint Satisfaction," In *Journal of Experimental Psychology: General*, 128, 3-31.

Hsiung, T. (1994), "Comment on a Wilcox Test Statistic for Comparing Means when Variances are Unequal," In *Journal of Educational Statistics*, 19, 2, pp. 111-18.

Inland Navigation Rules Act of 1980 (Public Law 96-591, 94 Stat. 3415, 33 U.S.C. 2001-2038).

International Navigational Rules Act of 1977 (Public Law 95-75, 91 Stat. 308, or 33 U.S.C. 1601-1608).

International Maritime Organization (2007), London, U.K.

IMO (2005) "Role of Human Element: Assessment of the Impact and Effectiveness of Implementation of the ISM Code," In *IMO - MSC 81/17/1, Maritime Safety Committee, 81st Session, Agenda Item 17*, 21 December 2005.

Iarossi, F. J. (2003), "Marine Safety: Perception and Reality," In *Proceedings of the 17th Annual Chua Chor Teck Memorial Lecture*, Singapore, 2003.

Jack, P.C. (1993), "Theory and Methodology Verification and Validation of Simulation Models," In *European Journal of Operational Research, 82*, pp. 145-162.

Jacko, J. A. and A. Sears (2003), "Fundamentals, Evolving Technologies and Emerging Applications," In *The Human-Computer Interaction Handbook*, Lawrence Erlbaum Associates, London, UK.

Jacobson, I., G. Booch and J. Rumbaugh (1999), *The Unified Software Development Process*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA.

Janis, I L. and L. Mann (1977), "Decision making," In *A Psychological Analysis of Conflict, Choice, and Commitment*, Free Press, New York, NY.

Johnson-Laird, P.N. (1993), *Mental Models*, Harvard University Press, Cambridge, MA.

Johnson-Laird, P.N. (1980), "Mental Models in Cognitive Science," In *Cognitive Science*, 4.1, pp. 71-115.

Joines, J. A., R.R. Barton, K. Kang and P.A. Fishwick (2000), "Experimental Design for Simulation," In *Proceedings of the 32nd conference on Winter Simulation*, Orlando, FL., pp. 32-38.

Joint Architecture for Unmanned Systems Compliance Specification (2006), Version 1.2, In *JAUS Working Group*, Pittsburg, PA.

Jonassen, D.H., M. Tessmer, and W.H. Hannum (1999), *Task Analysis Methods for Instructional Design*, Lawrence Erlbaum Associates, Mahwah, NJ.

Kahneman, D., A. Treisman, and B. J. Gibbs (1992), "The Reviewing of Object Files: Object-specific Integration of Information," In *Cognitive Psychology* 24, pp. 175-219.

Kamitani, Y. (2005), "Decoding the Perceptual and Subjective Contents of the Human Brain," In *Nature Neuroscience*, 8, pp. 679-685.

Kamitani, Y. (2005), "Decoding Seen and Attended Motion Directions from Activity in the Human Cortex," In *Current Biology,* 16, pp. 1096-1102.

Kelton, W. D. (2000), "Experimental Design for Simulation," In *Proceedings of the 2000 Winter Simulation Conference*, Orlando, FL., pp.32-38.

Kirchner, M. (2006), "Container Operations and Risk Handling from the Underwriter's Point of View," In *Proceedings of the International Union of Marine Insurance (IUMI) Conference, Tokyo*, Japan, 2006.

King, R., S. Diallo and A.Tolk (2007), "How to Play Fairly: Agents and Web Services Can Help," In *Proceedings of the Spring Simulation Interoperability Workshop*, Norfolk, VA., Simulation Interoperability Standards Organization, IEEE CS Press.

Klein, G. A. (1996), "The Development of Knowledge Elicitation Methods for Capturing Military Expertise," ARI Research Note 96-14, U.S. Army Research Institute for the Behavioral and Social Sciences, Alexandria, VA.

Klein, G. A., R. Calderwood and D. MacGregor (1989), "Critical Decision Method for Eliciting Knowledge," In *IEEE Transactions on Systems, Man, and Cybernetics*, 19.3, pp. 462-472.

Klimesch W. (1999), "EEG Alpha and Theta Oscillations Reflect Cognitive and Memory Performance: a Review and Analysis," In *Brain Res.,* 29, pp. 169–195.

Lackner, M.R. (1962), "Toward a General Simulation Capability," In *Proceedings of the AFIPS Spring Joint Computer Conference*, San Francisco, CA. pp. 1-14.

Laird, J.E. (2001), "Using Computer Games to Develop Advanced AI," In *Computer*, 34.7, pp. 70-75.

Lane, D.C., A. Größler and P.M. Milling (2003), "Rationality in System Dynamics: Modeling Human and Organizational Decision Making," In *Proceedings of the First European System Dynamics Workshop*, Mannheim University.

Larson, J., M. Bruch and J. Ebken (2006), "Autonomous Navigation and Obstacle Avoidance for Unmanned Surface Vehicles," In *SPIE Unmanned Systems Technology VIII*, Orlando, FL.

Law, A.M. and W.D. Kelton (2000), *Simulation Modeling and Analysis*, Third Edition,McGraw Hill, Boston, MA.

Likert, R. (1932), "A Technique for the Measurement of Attitudes," In *Archives of Psychology*, 140, p. 55.

Logothetis, N. K. and D.L. Sheinberg (1996), "Visual Object Recognition," In *Annual Rev. Neuroscience.*, 19, pp. 577-621.

Lowe, D.G. (1987), "The Viewpoint Consistency Constraint," *International Journal of Computer Vision*, 1, 1, pp. 57-72.

Lloyd's Register Fairplay (2005), *World Casualty Statistics*. Redhill: Lloyd's Register Fairplay, Ltd., London U.K.

Lloyd's Register Fairplay (2005), *World Fleet Statistics*. Redhill: Lloyd's Register Fairplay Ltd, London U.K.

Maida, A. S. (1990), "Frame Theory," *S. C.* Shapiro Ed., In *Encyclopedia of Artificial Intelligence*, Wiley, New York, NY, 1, pp. 302-312.

Merchant Shipping Act of 1976 (International Law).

Marocco D. and D. Floreano (2002), "Active Vision and Feature Selection in Evolutionary Behavioral Systems," In *Proceedings of the Seventh International Conference on Simulation of Adaptive Behavior; From Animals to Animats*, Edinburgh, UK, pp.247-255.

Marr, D. (1976), "Early Processing of Visual Information," In *Philosophical Transactions of the Royal Society of London*, Series B, 275, pp. 483-524.

Marr, D., and H. Ellen H. (1980), "Theory of Edge Detection," In *Proceedings of the Royal Society of London*, Series B, 207, pp. 187-217.

Marr, D. (1982), *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*, W.H. Freeman, San Francisco, CA.

Maya, V., M. Lamolle and C. Pelachaud (2004), "Influences and Embodied Conversational Agents Autonomous Agents and Multiagent Systems," In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2004)*," New York, NY. ACM Press, pp. 1306 – 1307.

McNeese, M. D. and M.A. Vidulich, Eds. (2002), "Cognitive Systems Engineering in Military Aviation Environments: Avoiding Cogminutia Fragmentosa," in *Technical Report, Human Systems Information Analysis Center*, Wright Patterson Air Force Base, OH.

Mernik, M., J. Heering and A. Sloane (2005), "When and How to Develop Domain-specific Languages," in *ACM Computing Surveys (CSUR) Archive*, 37, 4, pp. 316-344.

Merriam-Webster (2007), "Merriam Webster Online Dictionary," http://www.merriam-webster.com.

Metaxas, D. and D. Terzopoulos (1993), "Shape and Non-Rigid Motion Estimation through Physics-Based Synthesis," In *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15, 6, pp. 580-591.

Meyer, D.E. and D.E. Kieras (1995), "A Computational Theory of Human Multiple-Task Performance: Basic Mechanisms," In *Psychological Review*, 104, 1, pp. 3-65.

Meyer, G. M. and N.B. Szirbik (2007), "Agent Behavior Alignment: A Mechanism to Overcome Problems in Agent Interactions During Runtime", In *Cooperative Information Agents XI*, Vol. 4676, pp. 270-284.

Miller, J.A., G.T. Baramidze, P.S. Amit and P.A. Fishwick (2000), "Ontologies for Modeling and Simulation: An Extensible Framework," In *University of Florida ACM Journal*, Vol. V.

Miller H.J. (2004), "Tobler's First Law and Spatial Analysis," in *Annals of the Assoc. of American Geographers*, 94, 2, pp. 284-289.

Minsky, M. (1975) "A Framework for Representing Knowledge," In P. H. Winston (Ed.), *The Psychology of Computer Vision*, McGraw-Hill, New York, NY., pp. 211-277.

Militello, L. G. and R. J. B. Hutton (1998), "Applied Cognitive Task Analysis (ACTA): A Practitioner's Toolkit for Understanding Cognitive Task Demands," In *Ergonomics*, 41.11, pp.1618-1641.

Mitra, T. (1998), "On the Relationship between Discounting and Complicated Behavior in Dynamic Optimization Models," In *Journal of Economic Behavior and Organizations,* 33, 3-4 pp. 421-434.

Moya, L. J. and A. Tolk (2007), "Towards a Taxonomy of Agents and Multi-Agent Systems," In *Proceedings of the Agent Distributed Simulation ADS/ SCS Spring Simulation Conference*, Norfolk, VA.

Nash, J.F. (1950), "Equilibrium points in N-Person Games," In *Proceedings of the National Academy of Sciences of the United States of America*, 36, 1, pp. 48-49.

Neumann,V. J. and O. Morgenstern (1947), *The Theory of Games and Economic Behavior* 2nd edition, Princeton University Press, NJ.

Newell, A. (1990), *Unified Theories of Cognition,* Harvard University Press, Cambridge, MA.

Newton, Sir Isaac (Circa 1686), "Principia Mathematica Philosophiae Naturalis."

Nielsen, K. J., N. K. Logothetis and G. Rainer (2006), "Discrimination Strategies of Human and Monkey Observers for Complex Visual Displays," In *Current Biology*, 16, 8, pp. 814-820.

Oaksford, M., and N. Chater (1994), "A Rational Analysis of the Selection Task as Optimal Data Selection," In *Psychological Review*, 101, pp. 608-631.

Oaksford , M. and N. Chater (1998), *Rational Models of Cognition*, Oxford University Press, Oxford, U.K.

OptQuest™ Optimizer (2006), OptTek Systems, Inc., Boulder, CO.

Olson, G.M., and J.S. Olson (2003), "Human-Computer Interaction: Psychological Aspects of the Human Use of Computing," In *Annual Review of Psychology*, 54, pp. 491-516.

Ören, T. I., G.A.Nasser and L.Yilmaz (2007), "An Ontology-based Dictionary of Understanding as a Basis for Software Agents with Understanding Abilities," In *Proceedings of the Agent-Directed Simulation Symposium of the Spring Simulation Multi-conference (SMC'07)*, Norfolk, VA., pp. 19-27.

Ören, T. I. and L. Yilmaz (2004), "Behavioral Anticipation in Agent Simulation," In *Proceedings of the 2004 Winter Simulation Conference,* 1, pp. 806.

Pace, D. K. (2000), "Simulation Conceptual Modeling Issues," In *Proceedings of the 2000 Summer Computer Simulation Conference,* Interaction with Simulation Requirements, 2 , and Simulation Development Costs and V&V Costs , 3 , *Vancouver, BC.*

Pearl, J. (1988), *Probabilistic Reasoning in Intelligent Systems*, Morgan-Kaufmann, San Mateo, CA.

Pearl, J. (2000), *Causality: Models, Reasoning, and Inference*, Cambridge University Press, Cambridge, MA.

Pelechano, N., J.M. Allbeck and N. I. Badler (2007), "Behavior Modeling: Controlling Individual Agents in High-density Crowd Simulation," In *Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, Eurographics Association, San Diego, CA.

Peterson, M. A., B. de Gelder, S. Z. Rapcsak, P. C. Gerhardstein, and A. Bachoud-Levi (2000), "Object Memory Effects on Figure Assignment: Conscious Object Recognition is Not Necessary or Sufficient," In *Vision Research*, 40, pp.1549-1567.

Pew, R. W. and A.S. Mavor Eds.(1998), *Modeling Human and Organizational Behavior: Application to Military Simulations*, National Research Council, National Academy Press, Washington DC.

PIXAR Films, D. Baraff, A. Witkin and M. Kass (2003), "Untangling Cloth: ACM Transactions on Graphics," In *Proceedings of ACM SIGGRAPH*, 22, 3, pp. 862-870.

Platon, E., M. Mamei, N. Sabouret, S. Honiden and H. V. D. Parunak (2007), "Mechanisms for Environments in Multi-agent Systems: Survey and Opportunities," In *Journal of Autonomous Agents and Multi-Agent Systems*, 14, 1.

Pylyshyn, Z. (2002), "Mental imagery: In Search of a Theory," In *Behavioral and Brain Sciences*, Cambridge University Press, 25, 157-237.

Quest 3D® 3.0 VR Edition, product of Act-3D™ (2006), Leiden, The Netherlands.

Rational (2003), "Rational Unified Process and Unified Modeling Language," IBM, http://www.rational.com/.

Reynard, D., A. Wildenberg, A. Blake and J. Marchant (1996), "Learning Dynamics of Complex Motion from Image Sequences," In *Proceedings of the 4th European Conference on Computer Vision*, London, UK, pp. 357-368.

Riecken, D. (1994), "A Conversation with Marvin Minsky about Agents," In *Communications of the ACM Archive* , 37, 7, ACM Press New York, NY.

Riecke, B. E., M.V.D. Heyde and H.H. Bülthoff (2005), "Visual Cues Can be Sufficient for Triggering Automatic, Reflex-like Spatial Updating," In *ACM Transactions on Applied Perception* (TAP), 2 , 3, ACM Press, New York, NY, pp.183-215.

Riecke, B. E. and M. Von der Heyde (2002) "Qualitative Modeling of Spatial Orientation Processes Using Logical Propositions: Interconnecting Spatial Presence, Spatial Updating, Piloting, and Spatial Cognition," In *Tech. Report 100, MPI for Biological Cybernetics*, Max Planck Institute for Biological Cybernetics, Tübingen, Germany.

Rohr, K. (1994), "Towards Model-Based Recognition of Human Movements in Image Sequences," In *CVGIP: Image Understanding*, 59, 1, pp. 94-115.

Rouse, W. B. and N. M. Morris (1986), "On Looking Into the Black Box: Prospects and Limits in the Search for Mental Models," In *Psychological Bulletin*, 100(3), 349-363.

Rosen, R. (1985), *Anticipatory Systems: Philosophical, Mathematical and Methodological Foundations*, Pergamon Press, Oxford, U.K.

Russell S.and P. Norvig (2003), *Artificial Intelligence: A Modern Approach*, Prentice Hall, Englewood Cliffs, NJ.

SAS© Statistical Software (2007), SAS Institute, Cary, NC.

Safety of Life at SEA (SOLAS) International Treaty of 1974, Chapter V Regulation 19.2.1.4.

Savoy, R. L. (2001), "History and Future Directions of Human Brain Mapping and Functional Neuroimaging," In *Acta Psychologica*, 107, 9-42.

Schank R. and D. Leake (1989), "Creativity and Learning in a Case-based Explainer," *Artificial Intelligence*, 40, 1-3, pp. 353-385.

Schiller, P. (1996), "On the Specificity of Neurons and Visual Areas." In *Behavioral Brain Res.*,76, pp. 21-35.

Schraagen, J., S. Chipman and V. Shalin Eds. (2000), *Cognitive Task Analysis*, Lawrence Erlbaum, Mahwah, NJ.

Schrödinger, E. and W. Bitbol Ed. (1995), *The Interpretation of Quantum Mechanics*, Ox Bow Press, Woodbridge, CT.

Searl, J. (1969), *Speech Acts: An Essay in the Philosophy of Language*, Cambridge University Press, Cambridge, MA.

Silva, T., and O. Farias (2006), "Simulating Radiological Accidents through Software Agents," In *International Conference on Computational Intelligence for Modeling Control and Automation and International Conference on Intelligent Agents Web Technologies and International Commerce*, Sydney, Australia, pp. 96.

Silverman, B.G., M. Johns, J. Cornwell and K. O'Brien (2006), "Human Behavior Models for Agents in Simulators and Games: Part I: Enabling Science with PMFserv, Presence: Teleoperators and Virtual Environments," *MIT Press Journal*, 15, 2, pp.139-162.

Simon, H. A. (1990), "Prediction and Prescription in Systems Modeling," *Operations Research* 38(1), 7-14.

Simonin, O. and F. Gechter (2006), "An Environment-based Methodology to Design Reactive Multi-agent Systems for Problem Solving," In *Environments for Multi-Agent Systems II, LNAI 3830*, Springer, Berlin, pp. 32-49.

Sloman, S. A. (1996), "The Empirical Case for Two Systems of Reasoning," In *Psychological Bulletin,* 119, 3-22.

Sokolowski, J. (2003), "Enhanced Military Decision Modeling Using a MultiAgent System Approach," In *Proceedings of the Twelfth Conference on Behavior Representation in Modeling and Simulation*, Scottsdale, AZ., pp. 179-186.

Song, H., Z. Shen, C. Miao, A. Tan and G. Zhao (2007), "The Multi-Agent Data Collection in HLA-Based Simulation System," In *Proceedings of the 21st International Workshop on Principles of Advanced and Distributed Simulation,* San Deigo, CA. IEEE, pp. 61-69.

Sprague, N., D. Ballard and A. Robinson (2007), "Modeling Embodied Visual Behaviors," *ACM Transactions on Applied Perception (TAP)*, 4, 2, ACM Press, New York, NY.

Sterman, J.D. (2000), *Business Dynamics: Systems Thinking and Modeling for a Complex World,* Irwin McGraw-Hill, New York, NY.

Sternberg, S. (2001), "Separate Modifiability, Mental Modules, and the Use of Pure and Composite Measures to Reveal Them." In *Acta Psychologica*,106, pp.147-246.

Stewart, T.R., P. Middleton, M. Downton and D. Ely (1984), "Judgments of Photographs vs. Field Observations in Studies of Perception and Judgment of the Visual Environment," In *Journal of Environmental Psychology* 4, pp. 283-302.

Stormer, H. L., D.C. Tsui and A. C. Gossard (1999), "The Fractional Quantum Hall Effect," In *APS-Centennial Issue, Rev. Mod. Physics,* 71, pp. 298.

Tanaka, J., T. Curran and D.L. Sheinberg (2005), "The Training and Transfer of Real-world, Perceptual Expertise," In *Psychological Science*, 16, pp. 141-151.

Tanaka Y., N. Hayashibara , T. Enokido and M. Takizawa (2006), "A Fault-Tolerant Transactional Agent Model on Distributed Objects," In *Proceedings of the Ninth IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing* (ISORC'06), Gyeongju, KOREA, pp. 279-286.

Tanenhaus, M. K., M. J. Spivey-Knowlton, K. M. Eberhard, and J. C. Sedivy (1995), "Integration of Visual and Linguistic Information in Spoken Language Comprehension," In *Science*, 268, 1632-1634.

Taylor, R. M. (1989), "Situational Awareness Rating Technique (SART): The Development of a Tool for Aircrew Systems Design," In *Proceedings of the NATO Advisory Group for Aerospace Research and Development (AGARD) Situational Awareness in Aerospace Operations Symposium,* AGARD-CP-478.

Taylor, C. and D. Kriegman (1998), "Vision-based Motion Planning and Exploration for Mobile Robots," In *IEEE*, *Trans. on Robotics and Automation*, 14, 3, pp. 417-426.

Thorsden, M. (1991), "A Comparison of Two Tools for Cognitive Task Analysis: Concept Mapping and the Critical Decision Method," In *Proceedings of the Human Factors and Ergonomics Society, 35th Annual Meeting*, Santa Monica, CA.

Tolk, A. (1995), "Zur Reduktion struktureller Varianzen" (published in German; engl. Title: On the reduction of structural variances), Pro Universitate Verlag, Sinzheim.

Tolk, A. (2002), "Simulation of and for Military Decision Making," In *Proceedings of the RTO SAS Lecture Series*, The Hague, Netherlands, RTO-ENP-017.

Tolk, A (2005), "An Agent-based Decision Support System Architecture for the Military Domain," Chapter 8 in Gloria E. Phillips-Wren and C. J. Lakhmi, Eds., "Intelligent Decision Support Systems in Agent-Mediated Environments," In *Frontiers in Artificial Intelligence and Applications,* IOS Press, 115.

Tolk, A. and J. Garcia (2004), "Model-based Data Management for Mediation Services for Intelligent Software Agents," In *Symposium on Intelligent Software Systems for the New Infostructure, 16th International Conference on Systems Research, Informatics and Cybernetics,* Baden-Baden, Germany.

Tolk, A, S. Diallo and C. Turnitsa (2007), "Data, Models, Federations, Common Reference Models, and Model Theory," In *Proceedings of the European Simulation Interoperability Workshop: Simulation Interoperability Standards Organization*, ACM CS Press.

Troitzsch, K. and N. Gilbert (2004), "Simulation as a Tool to Model Stochastic Processes in Complex Socioeconomic Systems," In *Proceedings of the International Sociological Association (ISA) Research Committee on Logic and Methodology (RC 33) Tutorial 2004*, Amsterdam, Holland.

Turing, A. (1950), "Computing Machinery and Intelligence," *Mind* 59, 236, 433-60.

Uhrmacher, A. (2000), "A System Theoretic Approach to Constructing Test Beds for Multi-Agent Systems," In *A Tapestry of Systems and AI-Based Modeling and Simulation Theories and Methodologies: A Tribute to the 60th Birthday of B. P. Zeigler,* F. Cellier and H. Sarjoughian, Eds. Springer-Verlag, New York, NY.

U. C. G. Commandant (1972), "International Regulations for Prevention of Collisions at Sea", (COLREGS), US Department of Transportation, US Coast Guard, COMMANDANT INSTRUCTION M16672.2D.

Veit Elser, I. Rankenburg and P. Thibault (2007), "Searching with Iterated Maps," In *Proceedings of the National Acadamy of Sciences*, U S A., 104(2), pp.418–423.

Vanem, E., M. Eide and R. Skjong (2007), "Electronic Chart Display and Information Systems for Navigational Safety in Maritime Transportation," In *Proceeding of the E-navigation Conference*, October 2007, Oslo, Norway.

Von Bonin, G. and P. Bailey (1947), *The Neocortex of Macaca Mulatta*, University of Illinois Press, Urbana, IL.

VSTEP, Ship Simulator Software (2006), Rotterdam, The Netherlands.

Wan, K., V. Alagar and Z.Y. Yang (2007), "Trustable Ad Hoc Networks of Agent Societies," In *Proceedings of the Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing,* Qingdao, China, pp. 994-1000.

Wang, J. (2006), "Maritime Risk Assessment and its Current Status," *Quality and Reliability Engineering International*, Hoboken, NJ. 22:1, 3.

Watson, B., A. Friedman and A.McGaffey (2001), "Measuring and Predicting Visual Fidelity," In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, Los Angeles, CA. pp. 213-220.

Weisel, E. W., R. Mielke, and M. Petty (2003), "Validity of Models and Classes of Models in Semantic Composability", In *Proceedings of the Fall 2003 Simulation Interoperability Workshop*, Orlando FL, September , pp. 526-536.

Weixiong, Z. (2000), "A Template-Based and Pattern-Driven Approach to Situation Awareness and Assessment in Virtual Humans," In *Fourth International Conference on Autonomous Agents*, Barcelona, SP., pp. 116-123.

Wertheimer, M. (1922), "GestaltTheory, Translation by W. D. Ellis," *Source Book of Gestalt Psychology,* Harcourt, Brace and Co., New York, NY.

Wertheimer, M. and D. Beardslee (1958), "Principles of Perceptual Organization," In *Readings in Perception*, Princeton University Press, NJ, pp. 115-135.

Wickens, C. D., J.D. Lee, Y. Liu and B. Bordon (2004), *An Introduction to Human Factors Engineering*, Prentice-Hall, Upper Saddle River, NJ.

Xiufeng Z., J, Yicheng, Y. Yong and L. Zhihua (2004) , "Ship Simulation using Virtual Reality Technique," In *Source Proceedings of the 2004 ACM SIGGRAPH International Conference on Virtual Reality Continuum and its Applications in Industry*, Singapore, 5, pp. 282- 285.

Yaari, A. (2008), "Implementing MOOS and IvPHelm on a USV," In *Proceedings of the AUVSI Annual Symposium for Unmanned Systems North America*, June 11-12.

Yantis, S. (1992), "Multi-element Visual Tracking: Attention and Perceptual Organization," In *Cognitive Psychology*, 24, pp. 295-340.

Yeh, E. and D. Kriegman (1995), "Toward Selecting and Recognizing Natural Landmarks," In *IEEE International Workshop on Intelligent Robots and Systems*, Nagoya, Japan, pp. 47-53.

Ye Y., S. Boies, P. Huang and J. Tsotsos J. (2001), "Smart Distance and WWWaware – A Multi-Agent Approach," In *Proceedings of the Fifth International Conference on Autonomous Agents*, Montreal, Quebec, pp. 176-177.

Yilmaz L. (2006), "Validation and Verification of Social Processes within Agent-Based Computational Organization Models," In *Computational and Mathematical Organization Theory*, 12, 4, pp. 283-312.

Yilmaz, L., T. Ören, and A. Nasser-Ghasem (2006), "Agents, Simulation, and Gaming," *Simulation and Gaming Journal*, 37, 3, pp. 339-349.

Yilmaz, L. and T. Ören, Eds. (2005), "Agent-Directed Simulation," In *Special Issue of the Transactions of the Society for Computer Simulation International*, 81, 7.

Zabinsky, B. Zelda and Robert L. Smith (1992), "Pure Adaptive Search in Global Optimization," In *Mathematical Programming,* 55, pp. 323-38.

Zeigler, B.P., H. Praehofer and T. Kim (2000), "Integrating Discrete Event and Continuous Complex Dynamic Systems, *"Theory of Modeling and Simulation"*, Second Edition, Academic Press, Orlando, FL.

Zeigler, B.P. (1986), "Systems Knowledge: A Definition and its Implications," In: *Modeling and Simulation Methodology in the Artificial Intelligence,* Era, M.S, Elzas, Ören, T.I., and B.P Zeigler Eds., North-Holland, Amsterdam, pp. 15-17.

APPENDIX

Supporting information is listed is an orderly categorization used in introducing AP to improve HBR. These excepted psychology practices are used to extract knowledge that comprise cognitive elements representing agent decision making solutions. This study demonstrates a valid approach to measuring perception using agent solutions and observations that these expert solutions outperform state-of-the-art solutions.

Figure A-1 is a knowledge questionnaire that helps identify expert working knowledge associated with a critical ship navigation task.

Figure A-1. SME Qualification Questionaire

SME QUALIFICATION QUESTIONNAIRE

Are you, or have you been, a ship conning officer or operator directly responsible for ship maneuvering in a professional capacity? Y/N

Please list the types of ships you have navigated:

Ships	**Organization**	**Types of duties or responsibilities related to ship navigation**

Would you, or have you ever considered yourself a proficient ship handler? Y/N

How many years/months experience have you had in ship handling? ____________

Regarding this experience, how many of these years/months involved critical ship navigation in restricted harbors or waterways. _____________

Age ______ Sex M/F_______ Comments:

For assessing the perceptual skills associated with a critical task, a knowledge audit worksheet is completed by SMEs as shown in figure A-2.

Figure A-2. Knowledge Audit Worksheet

KNOWLEDGE AUDIT WORKSHEET

Task: Ship maneuvering in a restricted channel

1. Perceptual skills: (identify cues and patterns that a novice would not detect; list examples. For each cue and strategy, why would it be hard for a novice to perform effectively?)

Expertise (examples)	**Cues and Strategies**	**What makes this task difficult?**

2. Past and future: (Is there a time when you walked into the middle of a situation and knew exactly how things got there and where they were headed?)

Expertise (examples)	**Cues and Strategies**	**What makes this task difficult?**

3. Mental Model ("big picture"): (Give an example of what is important about the Big Picture for this task: What are the major elements you have to know and keep track of?)

Expertise (examples)	**Cues and Strategies**	**What makes this task difficult?**

4. General rules of thumb when in restricted waters: (When performing this task are there ways of working smart or accomplishing more with less that you have found especially useful?)

Figure A-2, Continued

Expertise (examples)	**Cues and Strategies**	**What makes this task difficult?**

5. Improvising: (Are there any examples of when you improvised in this task, or noticed an opportunity to do something more quickly or better and followed up on it?)

Expertise (examples)	**Cues and Strategies**	**What makes this task difficult?**

6. How do you self-monitor performing this task: (Most experienced people when performing a critical task check their performance and make necessary adjustments. Can you think of any examples where you have done this in operating a vessel in restricted navigation?)

Expertise (examples)	**Cues and Strategies**	**What makes this task difficult?**

7. Anomalies: (Can you describe an instance when you spotted a deviation from the norm, or knew something was amiss? Have there been times when the equipment pointed in one direction, but your own judgment told you to do something else? Alternatively, when you had to rely on experience to avoid being led astray by the equipment?)

Expertise (examples)	**Cues and Strategies**	**What makes this task difficult?**

Table A-1 is the knowledge audit interview listing the basic probes for the task associated with avoiding a tug pushing a container barge in a restricted waterway.

Table 25. Knowledge Audit Interview Probes

TUG PUSHING CONTAINER BARGE IN RESTRICTED WATERWAY

Basic Probes

Perceptual Skill	Detect cues and patterns and make discriminations that novices cannot see. List examples:
Past & Future	Have you experienced a tug pushing container barge in restricted waterway and knew exactly how they were positioned and what my happen in the near term future?
Big Picture	What is the "Big Picture" for this task: What are the major elements you have to know and keep track of while maneuvering to avoid the tug?
Job Smarts	While maneuvering to avoid the tug, are there methods you have found especially useful?
Opportunities / Improvising	Are there examples of improvising with this task?
Self Monitoring	How do you check your performance and make adjustments, if necessary, to avoid the tug. Any specific examples of this?
Anomalies	Have you experienced an instance when you spotted a deviation from the norm, or sensed something was wrong?
Equipment Difficulties	Have there been times when the navigation aids in one direction, but your own judgment told you needed to react differently or you had to rely on experience to avoid being led astray by the navigation aids?

Table 26. Critical Navigation Visual Cues from Simulation Interviews

	SIMULATION INTERVIEW CRITICAL CUES
Events	Tug pushing container barge and crossing situation where both vessels are moving based on geographic constraints.
Actions	Determine anticipated direction and speed of tug and container barge. Would maneuver as necessary based on tug and container barge movements. Would have contacted port operations for other scheduled ships getting underway. Will slow down in speed to further assess if there will be room to turn in desired direction. Will turn own ship if necessary. Will stop own ship if necessary. If necessary will go max speed astern to avoid collision.
Assessment	Will ascertain if approaching tug is aware of my presence (Do they see me?) Own ship set and drift. Will assess estimated time and range of contact for crossing vessels. Will determine if we are in an extreme situation and if action is needed to avoid collision.
Critical Cues	If visual only and unable to call and determine intentions, will note any flag hoists or day shapes that may assist in the assessment.

	Will view container barge and tug profiles to estimate bearing and distance.
	Will note bearing drift to determine direction.
	Will observe wake of container barge and estimate speed.
	Will observe closing range of vessels.
	Course deviation.
	Flow of water around buoys.
	Noticeable wave action.
Potential Errors	Unable to determine whether other vessels are underway.
	Misjudgment in tug and container barge bearing and range.
	Miscalculation of winds and tidal currents.
	Error in reacting to situations outside the "rules of the road".
	Error in judgments on anticipated tug and container barge movement.
	Wrong navigation action if in extremis.

Table 27. Knowledge Audit Interview Task

TUG PUSHING CONTAINER BARGE IN RESTRICTED WATERWAY

Aspects of Expertise	Cues & Strategies	Why Difficult
Perceptual Skills:		
Current determination and effects of wind	Own ship set and drift Observation of water flow around buoys Observation of wave action	Subtle effects of these may be unnoticeable
Bearing and drift	Use of ship navigation sensors and experience to determine	Bearing drift sometimes is less noticeable due to own ship motion and movement. Also can look different at differing times of day
Target Angle	Constant observation required; observe bow wake to see if forward part of target aspect	With changing distance, difficulties include ascertaining topside configuration. Also, difficult if multiple contacts and objects are in close proximity to each other
Container barges/Tugs	Direction of movement Bow wake (white water)	Tug pushing container barge - cannot be seen behind container barge and hard to determine intentions/movement Tug towing - hard to see tow line
Submarines	Direction of movement	Very small profile in comparison to size of ship
Ships at anchor	Lack of wake	Hard to see anchor chain
Effect of current	Cues based on research	Specific effect of currents

Table A-3, Continued

Aspects of Expertise	Cues & Strategies	Why Difficult
Ships and container barges	Some can observe target angle. Some observed through flying signals/radio	Local arrangements are unique and may require customized response
Past and future:		
Converging contacts with own ship where situation is not covered by "rules of the road"	Slow speed of advance Gain inaccurate maneuvering board solution on each contact Unable to contact via UHF Sound signals	Not specifically covered by rules of the road protocol
Crossing situation where the stand-on vessel maneuvered too early	Observed that protocols have been violated – must watch closely	Unsure "who expects who" is to maneuver, once protocols are violated
Mental Model (big picture):		
Relative motion	Visual Observation	Potential distractions to navigator and watch team
Radar picture inaccurate	Visual Observation	Potential distractions to navigator and watch team
Whether other ship watches are aware of influencing situation	Visual Observation	Potential distractions to navigator and watch team
Hazards to navigation	Visual Observation	Potential distractions to navigator and watch team
Traffic scheme, buoys, ranges, and other aids	Visual Observation	Potential distractions to navigator and watch team

Table A-3, Continued

Aspects of Expertise	Cues & Strategies	Why Difficult
Weather	Visual Observation	Potential distractions to navigator and watch team
Anticipating movement of ships	Position of ships	Unpredictable movement
	Position of tugs	Unpredictable movement
	Position of buoys	Unpredictable movement
	Position of container barges	Unpredictable movement
Advance knowledge of where a ship is going such as call ahead	Knowing where certain types of ships are located (e.g., container ships, off load container barges)	Unpredictable ship awareness
Determining what contact comes into play next	Strategies include planning ahead for many ships moving about in crowded waterway	May not know what ships will interact with until the near future situation develops
General rules of thumb when in restricted waters:		
Fully adhere to the "rules of the road"	NA	Less watch standers means less eyes looking 360 degrees
Communicate in advance with the other approaching ships		
Locate working tugs	Tugs will "preposition" at turns in a channel ahead of large ships that may require assistance in maneuvering or making a turn	Not difficult if experienced navigator and know what to look for

Table A-3, Continued

Aspects of Expertise	**Cues & Strategies**	**Why Difficult**
Law of gross tonnage	Small boats need to stay clear of large vessels despite "rules of the road"	People do stupid things like running in front of large ships
Keep experienced navigator watching the "unassigned" ships	Keep bridge personnel informed	Not difficult if experienced
For a large vessel, make good communications with bow and stern watch standers	Keep bridge personnel informed	Not difficult if experienced
Have harbor common communications/VTC net up on bridge	Keep bridge personnel informed	Not difficult if experienced
Improvising:		
In restricted visibility such as severe fog, send personnel to bow with radio to listen for contacts and buoys	Bells from buoys Sounds made by approaching ships	Unaware of potential contacts
In un-chartered waters, send personnel to bow with depth finder	Strategies will depend on situation encountered	No soundings or charts
One example, had to turn around and head the opposite direction to ensure that if they hit, they would have to overtake ship. (Event actually Occurred)	No cues or strategies	Unplanned
		Not predictable

Table A-3, Continued

Aspects of Expertise	Cues & Strategies	Why Difficult
Improvised by using navigable waters outside the buoys for a separation scheme to avoid collision	Departure from “rules of the road”	Taught to trust navigation aids
Use of “Seaman’s eye” to navigate	Unsure of navigation aids	
Improvising	Not good strategy	Not recommended
How do you self monitor performing this task?		
Having a watch section providing additional information to the bridge	Cues provided by watch standers	Time consuming
Observe change in bearing	Watch how a contact’s bearing changes to determine if it will cross ahead or astern, or if it is on a collision course	Many people do not pay attention to relative bearing Relative bearing and bearing rate will change any maneuver by own ship or contact May not have complete navigation picture
By “prebriefing” prior to navigation into restricted waters	Study charts and obtain expected traffic information	Easy not to take the time and log the information following the event unless there was an incident
By “debriefing” and logging results		
Anomalies:		
Maneuvering at night; navigation running lights can be deceiving with regard to speed	*Own ship maneuver results in much higher bearing rate change that expected*	*Night maneuvering is very difficult on perceptions of oncoming ships*

Table A-3, Continued

Aspects of Expertise	**Cues & Strategies**	**Why Difficult**
Not anticipating on how fast a ship is moving and cutting in front of it	*Not paying attention to the size of bow wave which can give an indication of speed*	*Determining ship speed*
Loss of gyro	*Notice gyro lost*	*Used for bearing information*
Sailing in restricted waters in low visibility	*Can't see contacts*	*Unable to assess oncoming ships movement*

Figure A-3. Simulation Interview Worksheet

SIMULATION INTERVIEW WORKSHEET

Simulation: Ship Maneuvering in a Restricted Channel

Please review the ship navigation simulation. This simulation presents a critical maneuver that has been identified in the knowledge audit.

Events	**Comments**
Think about this scenario. Please list the primary events that occurred during the incident. List events that include judgments or decisions points.	
Actions	
What actions, if any, would you take?	**Comments**
Situation Assessment	**Comments**
Please list what you think is going on here? What is your assessment of the situation at this point in time?	
Critical Cues	
Please identify the pieces of information that led you to this situational assessment and related actions.	**Comments**
Potential Errors	
What errors would an inexperienced person be likely to make in this situation?	**Comments**

Table 28. Sampled Simulation Interview Probe Results

CRITICAL NAVIGATION ACTION HIERARCHY SAMPLED RESULTS FROM

SIMULATION PROBE

Events	*Actions*	*Assessment*	*Critical cues*	*Potential errors*
Initial visual contact with Tug and Container ship	Determine own ship frame of reference	Is it underway? Will each cross navigation paths? Threat?	Bow angle of contact	Not aware of tug's path or own ship's course-judgment errors.
Notice tug is pushing the Container ship?	Must give way if a burdened vessel	Look for signs of burdened vessel	Bow wake of Container ship and possible flag hoists or horn blasts	Not allow time to maneuver away from Container ship
Not able to determine Container ship/tug navigational track	Determine burden ship's course, speed and anticipated future location	Ascertain through plotting own ship's relation to burdened vessel	Constant target angle with own ship's course and speed - visual movement of Container ship	Ship collision

Figure A-4. Cognitive Graph for Critical Navigation Action Hierarchy

CRITICAL NAVIGATION ACTION HIERARCHY

The cognitive graph is used to help identify important cues that can be mapped to UML for building an agent representation.

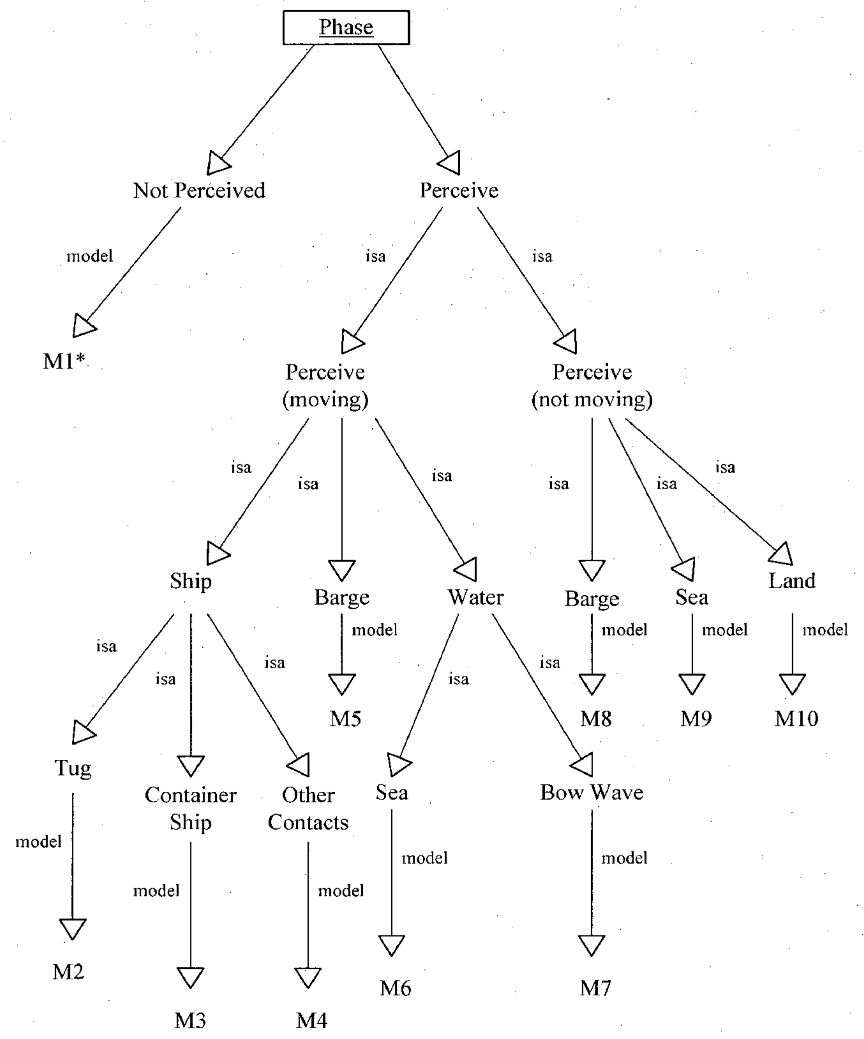


Figure A-5. Mapping UML to Meta-Model

CRITICAL NAVIGATION META-MODEL

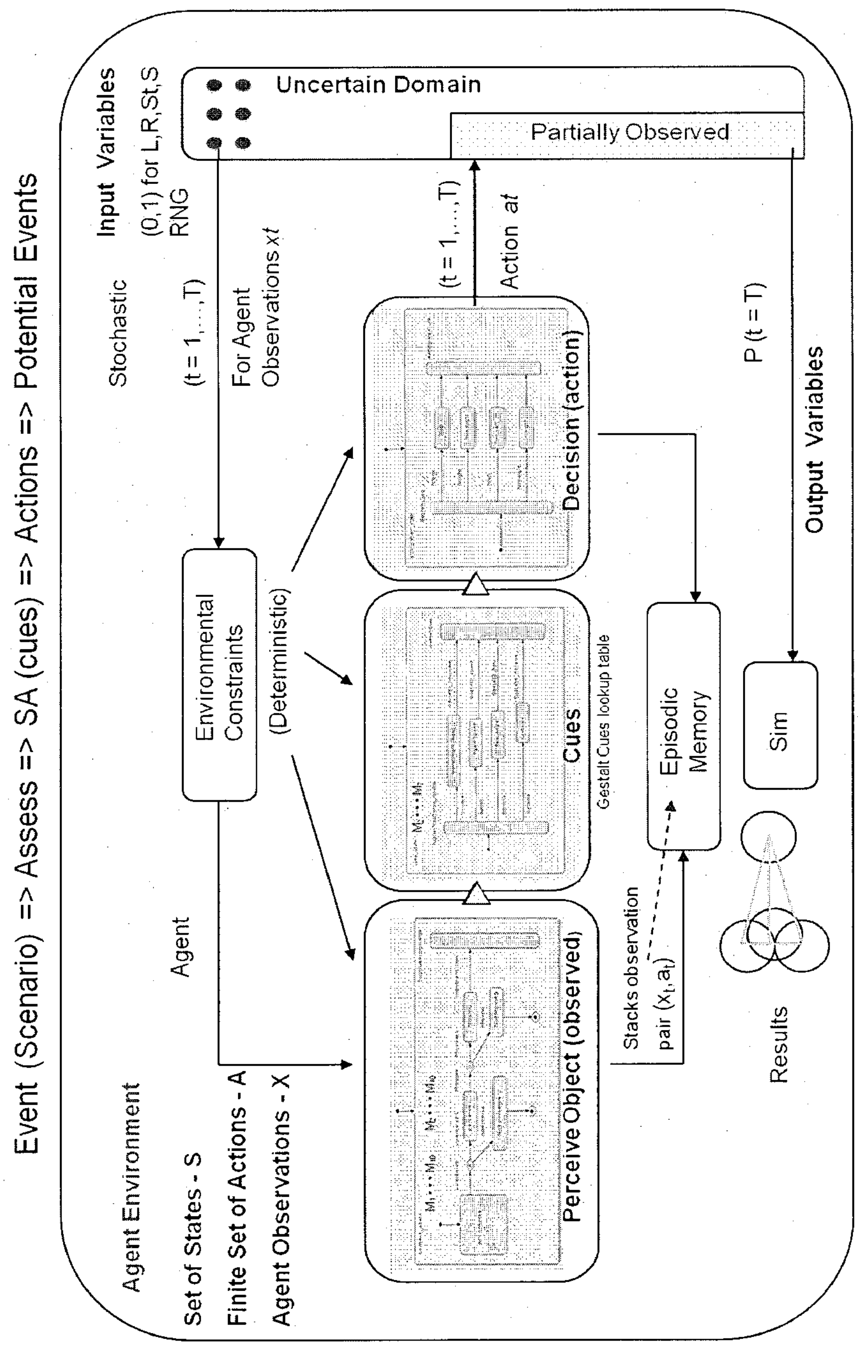
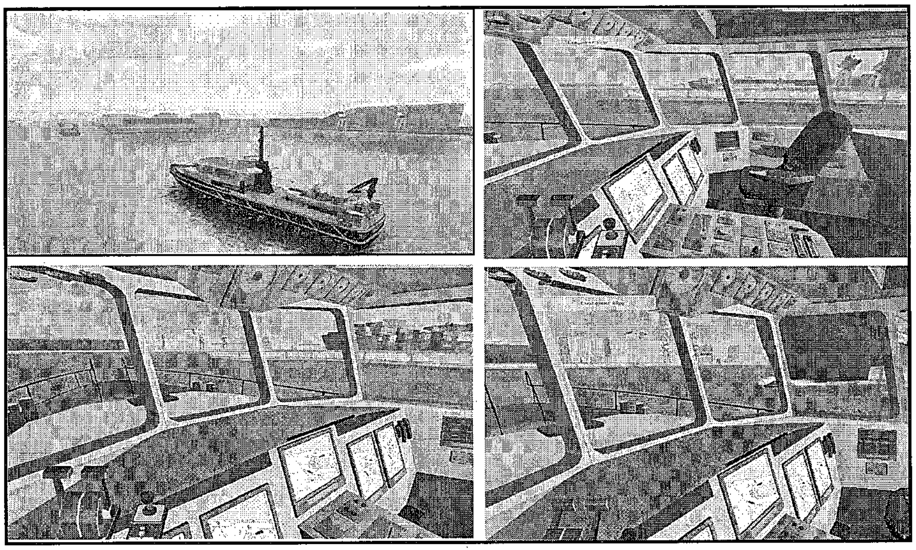
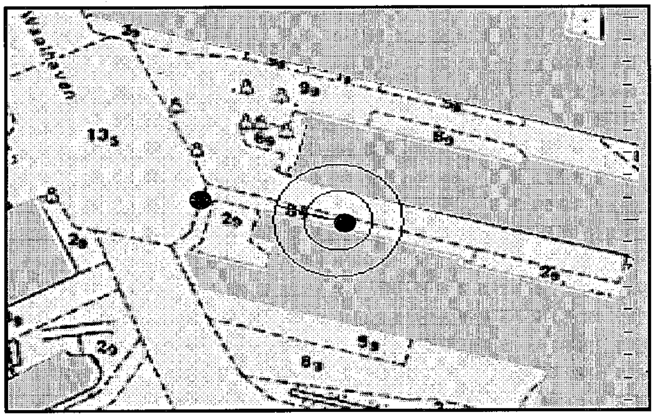


Figure A-6. Sampled Screen Shots from Simulation Probes

SAMPLED SCREEN SHOTS FROM SIMULATION PROBES



Various Test Scenario Views from the Bridge



VSTEP, Ship Simulator Software Chart of a Tug Pushing Container Ship in a Restricted Waterway

Figure A-7. Schematic Description of Agent Perception

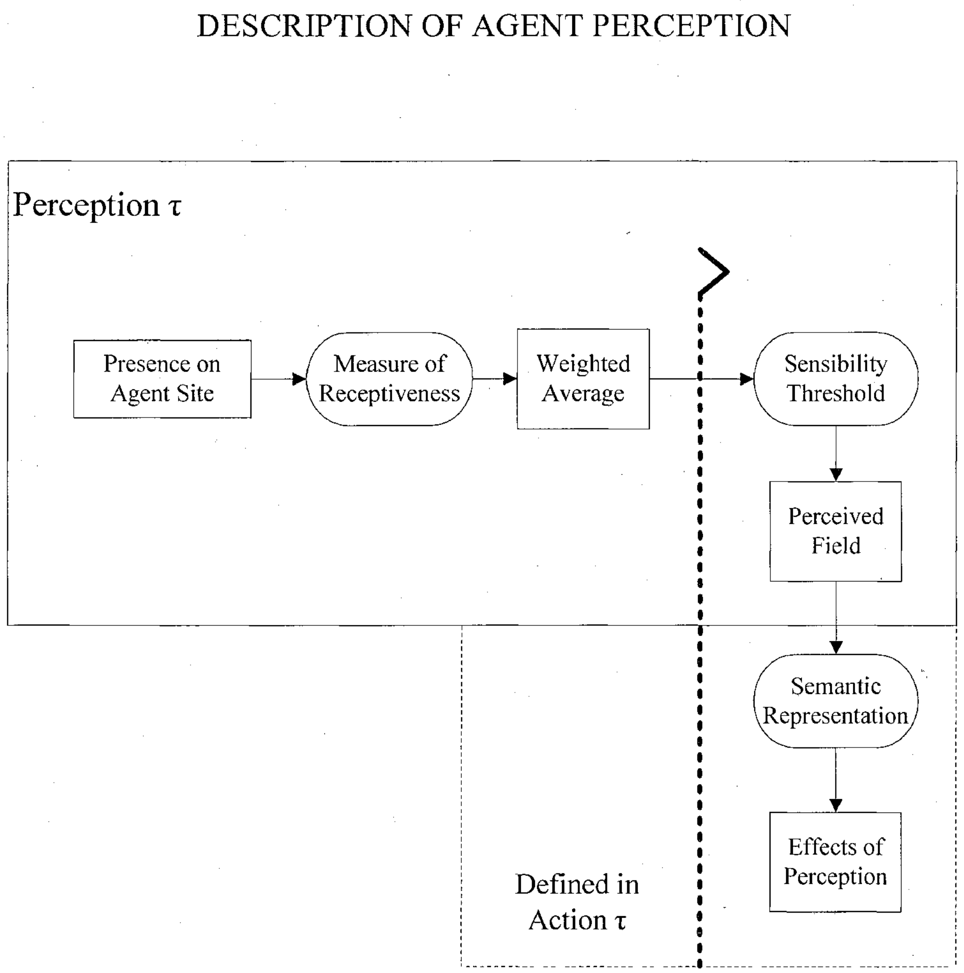


Figure A-8. HBR AP Logic Validation Overlay

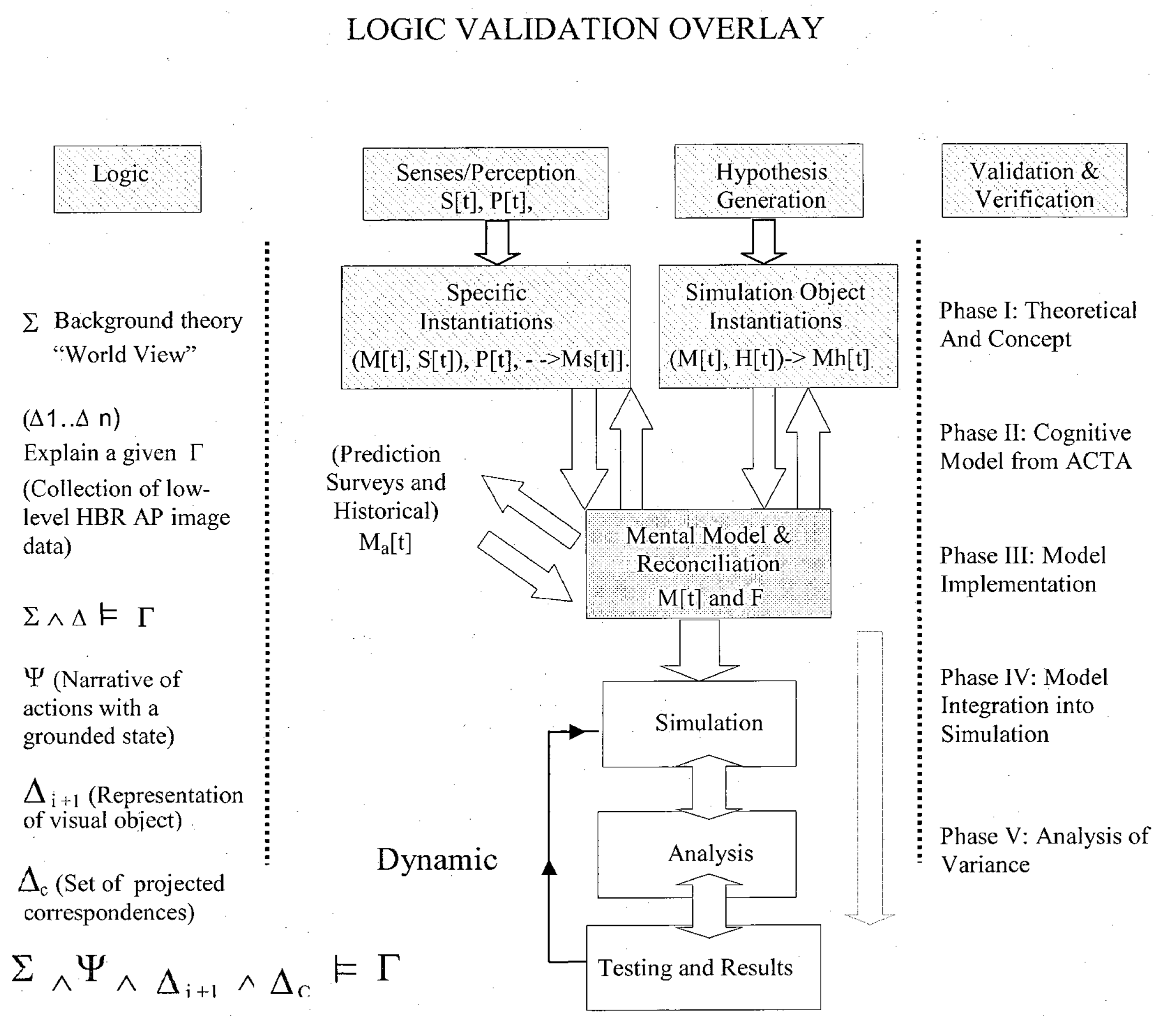


Figure A-9. Conceptual Graph Analysis Figure

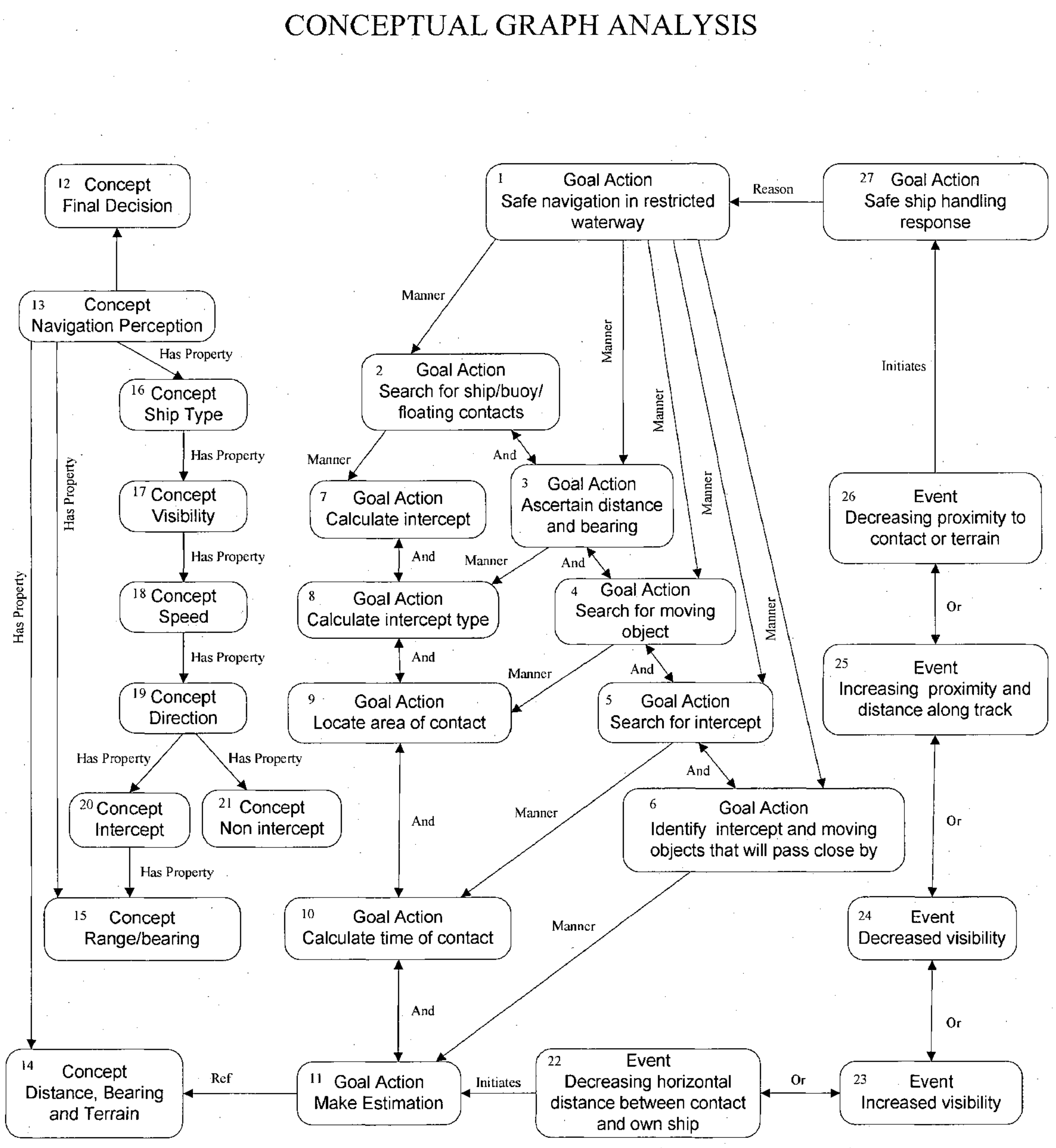
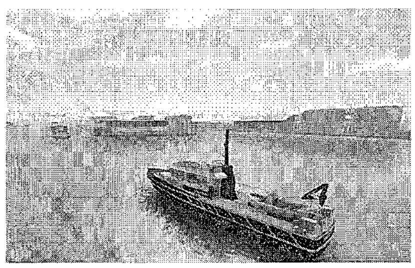
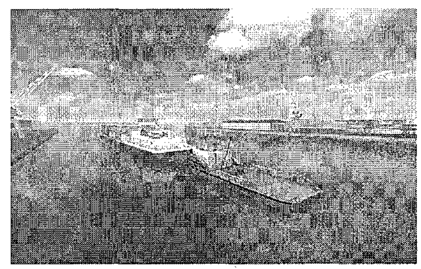


Figure A-10. Critical Task Scenario Image

CRITICAL TASK SCENARIO GRAPH

Simulation gaming software replicates the critical task scenario (tug pushing container barge in restricted waterway)





Simulation Interview Probes/Results

Events	Actions	Assessment	Critical cues	Potential errors

Events from critical task e.g. restricted navigation task scored on a Likert scale.

Figure A-11. AP Simulation Model Variables and Properties

AP SIMULATION MODEL INPUT PARAMETERS

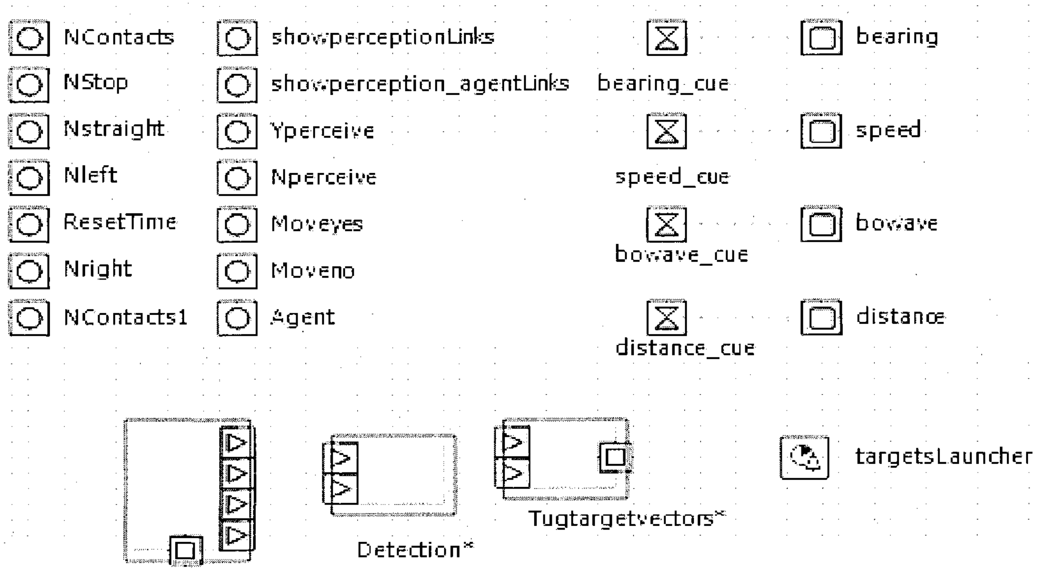
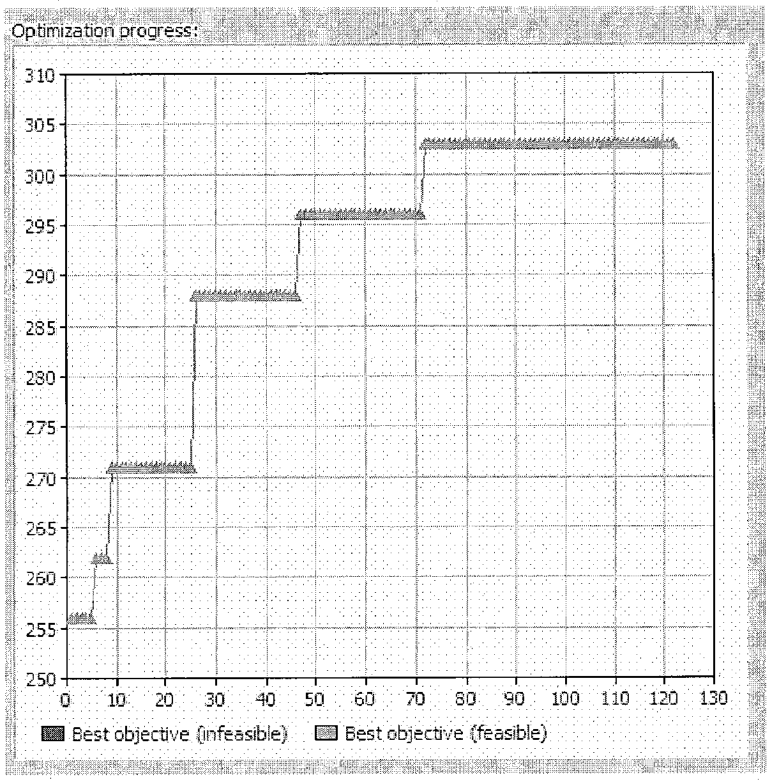
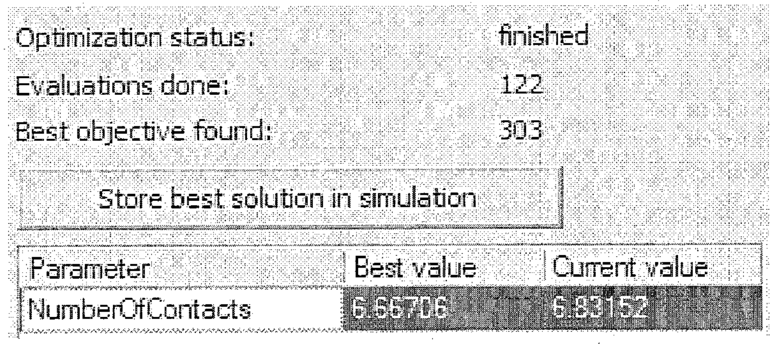


Figure A-12. AP Simulation Parameter Optimization

AP MODEL PARAMETERIZATION OPTIMIZATION





A. EXPERIMENT RESULTS

Experimental results demonstrate how to apply an AP approach using proven psychological methods. Figure A-13 shows and overlay plot of agent and expert means based on an anticipated response.

Figure A-13. Overlay Plot of $meanAgent_{random}$ and Expert meanAgent by Anticipated Response

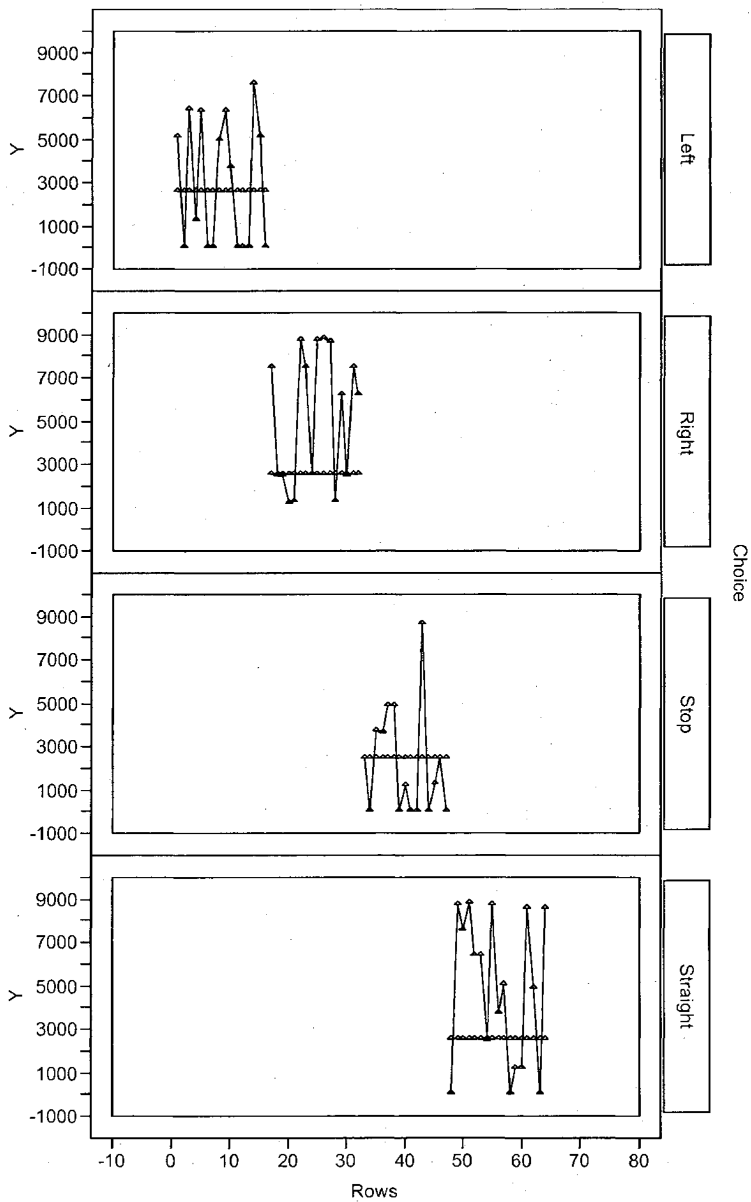


Table 29. Covariance and Partial Correlation of Expert Agent vs. Random Agent

The main effect (marginal) means shown as observed through partial correlation:

P.Cov	Agent Seeded	Agent Random
Agent Seeded	10502722.1	15068.4464
Agent Random	15068.4464	2632.74603

P.Corr	Agent Seeded	Agent Random
Agent Seeded	1.0000	0.0906
Agent Random	0.0906	1.0000

Overall, subjects tended to select choices differently from agent random agent selection. Random selection Tables A-6 through A-8 provide observation of actual random selection results:

Table 30. Observation of ANOVA Object Random Summary of Fit

Rsquare	1
Adj Rsquare	1
Root Mean Square Error	0
Mean of Response	2522.625
Observations	64

Table 31. Observation of agent Random Mean Square and Sum of Squares

Source	DF	Sum of Squares	Mean Square
Choice	3	165863.00	55287.7
Error	60	0.00	0.0
C. Total	63	165863.00	

Table 32. Observation of Agent-Object Random Selection by Choice

Level	Number	Mean	Lower 95%	Upper 95%
Left	16	2590.00	2590.0	2590.0
Right	16	2543.00	2543.0	2543.0
Stop	15	2449.00	2449.0	2449.0
Straight	17	2505.00	2505.0	2505.0

Figure A-14 shows the response variable Y as a numeric continuous variable and the X variable, Agent Random Mean Choice, as a nominal (categorical) variable. The fit Y by X data in table 8 for the analysis used SAS-JMP, software version 6.0 [SAS© 2007] Table A-9 shows the mean and standard deviation.

Figure A-14. Visual Comparison of Group Means Side-by-Side Vertical Scatter Plot

Fit X by Y Analysis:

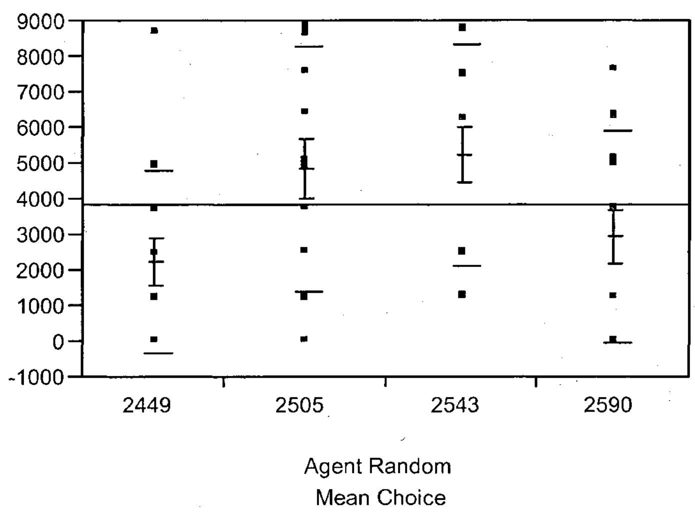


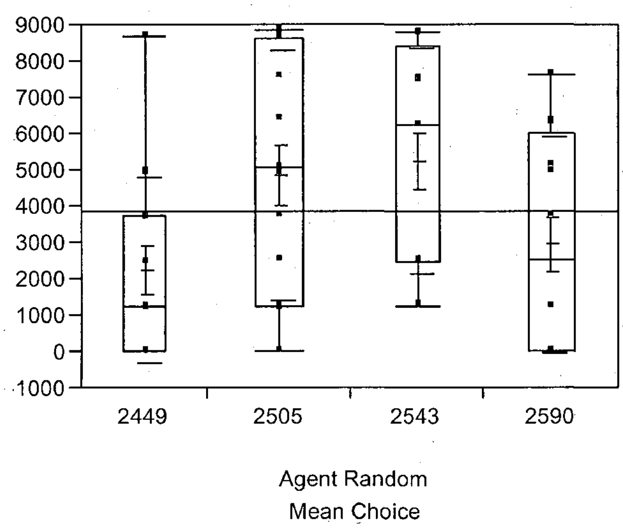
Table 33. Means and Standard Deviation Data for XY Fit Model

Random Mean	**Choice**	**Mean**	**Std Dev**	**Std Err Mean**	**Lower 95%**	**Upper 95%**
2449	Stop	2216.73	2569.13	663.35	794.0	3639.5
2505	Straight	4849.18	3447.56	836.16	3076.6	6621.7
2543	Right	5223.50	3105.02	776.26	3568.9	6878.1
2590	Left	2919.75	2982.89	745.72	1330.3	4509.2

The percentile plots below divide the data so that n% of the data are equal to or below the nth quartile.

Figure A-15. Means Fit Y by X Quartile Plot

Figure A-15 illustrates data shape and symmetry:



The horizontal line inside the quartile box plot represents the median, e.g., 50th quartile. Half the values are at or below the 50th quartile and half are above. The top and bottom of the box represent the 25th and the 75th quartiles. "Whiskers" at both ends of the box extend from the end of the box to the outer-most data point that falls within 1.5 times the range from the 25th to the 75th quartile.

Figure A-16. Results of Agent-Object Selection by Choice

Figure A-16 provides graphic observation of actual random selection results:

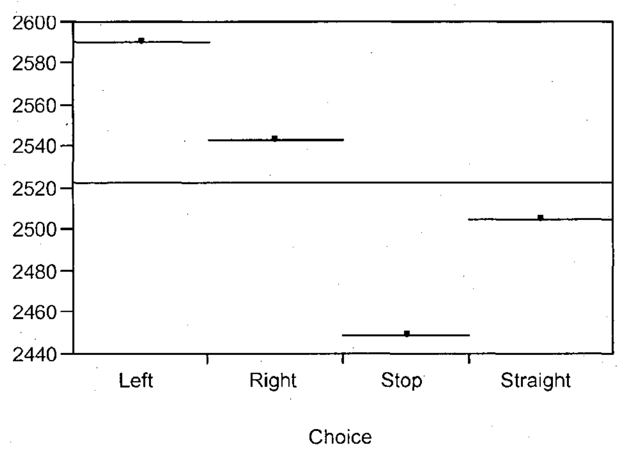


Table A-10 presents an example of how to observe actual random selection results. By injecting anticipation variables that are equal to Agent subjects tended to select perception choices different from the standard meanAgent = 3843.422 selection. It provides observation of actual agent results:

Table 34. Observation of Agent Results

ANOVA
Summary of Fit

Rsquare	0.152663
Adj Rsquare	0.110296
Root Mean Square Error	3056.847
Mean of Response	3843.422
Observations (or Sum Wgts)	64

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Choice	3	101012519	33670840	3.6033	0.0184
Error	60	560658974	9344316.2		
C. Total	63	661671494			

Means for ANOVA

Level	Number	Mean	Std Error	Lower 95%	Upper 95%
Left	16	2919.75	764.21	1391.1	4448.4
Right	16	5223.50	764.21	3694.8	6752.2
Stop	15	2216.73	789.27	637.9	3795.5
Straight	17	4849.18	741.39	3366.2	6332.2

Std Error uses a pooled estimate of error variance

Table A-10, Continued

Means and Standard Deviations

Level	Number	Mean	Std Dev	Std Err Mean	Lower 95%	Upper 95%
Left	16	2919.75	2982.89	745.72	1330.3	4509.2
Right	16	5223.50	3105.02	776.26	3568.9	6878.1
Stop	15	2216.73	2569.13	663.35	794.0	3639.5
Straight	17	4849.18	3447.56	836.16	3076.6	6621.7

For simulation values, on average, across all conditions, subjects believed more frequently that the ship would turn right $\text{Mean}_{\text{right}}$ = 5223.50 or continue straight $\text{Mean}_{\text{straight}}$ = 4849.18 (Mean of response = 3843.422). Anticipation variables appeared to positively influence Agent. Figures A-17 and A-18 are graphic observations of actual agent results. Table A-11 provides a quartile plot for the agent data:

Figure A-17. Graphic results of AP Object Values by Choice

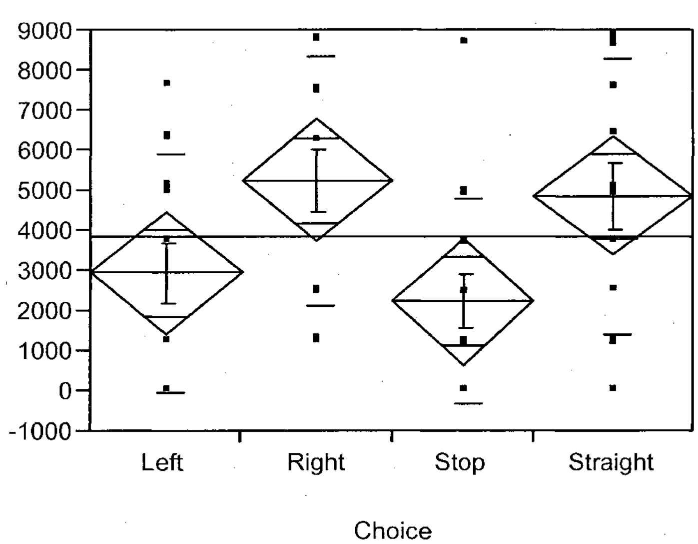


Figure A-18. Quantile Plot Agent versus Random Mean Choice

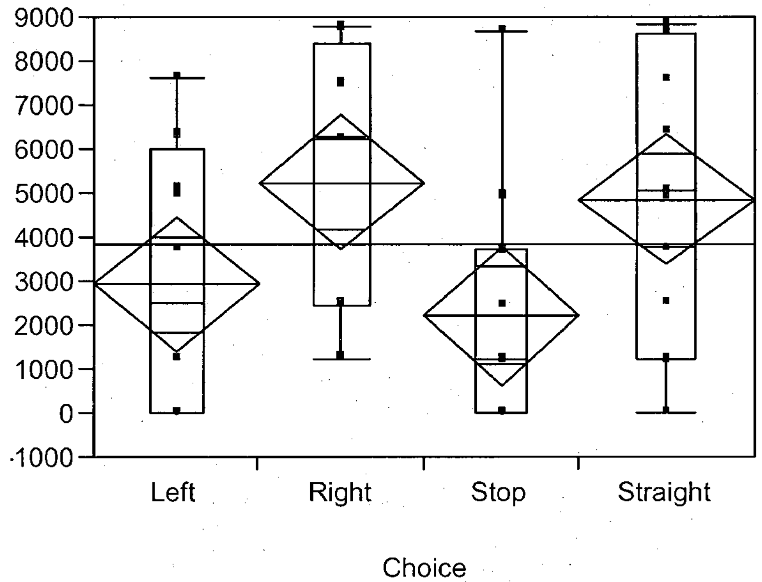


Table 35. Quartile Plot Agent Data

Quartiles

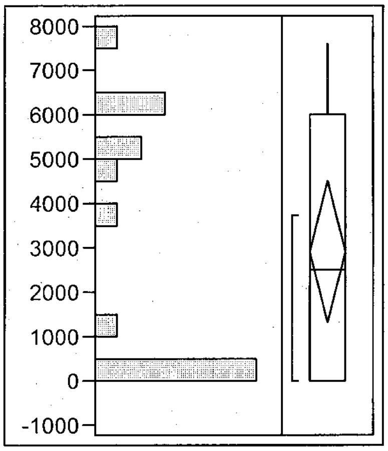
Level	Minimum	10%	25%	Median	75%	90%	Maximum
Left	0	0	0	2490	6002	6728.2	7592
Right	1223	1241.2	2456.75	6218	8406.25	8781.8	8800
Stop	0	0	0	1243	3748	6426.2	8687
Straight	0	0	1207	5067	8610	8749.4	8819

Quartile Plot Random

Level	Minimum	10%	25%	Median	75%	90%	Maximum
Left	2590	2590	2590	2590	2590	2590	2590
Right	2543	2543	2543	2543	2543	2543	2543
Stop	2449	2449	2449	2449	2449	2449	2449
Straight	2505	2505	2505	2505	2505	2505	2505

Figure A-19. Graphic Displays of Quartile Charts and Data

Agent



Choice=Left

Quartiles

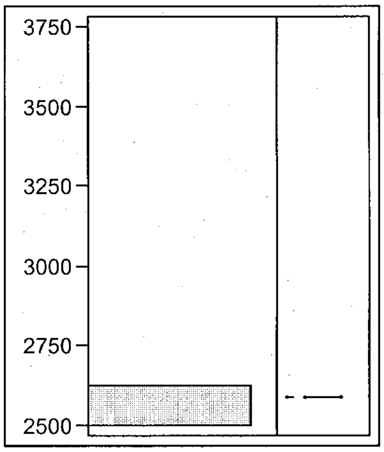
100.0%	maximum	7592.0
99.5%		7592.0
97.5%		7592.0
90.0%		6728.2
75.0%	quartile	6002.0
50.0%	median	2490.0
25.0%	quartile	0.0
10.0%		0.0
2.5%		0.0
0.5%		0.0
0.0%	minimum	0.0

Moments

Mean	2919.75
Std Dev	2982.8933
Std Err Mean	745.72333
upper 95% Mean	4509.2217
lower 95% Mean	1330.2783
N	16

Distributions

Agent Random



Random

Quartiles

100.0%	maximum	2590.0
99.5%		2590.0
97.5%		2590.0
90.0%		2590.0
75.0%	quartile	2590.0
50.0%	median	2590.0
25.0%	quartile	2590.0
10.0%		2590.0
2.5%		2590.0
0.5%		2590.0
0.0%	minimum	2590.0

Moments

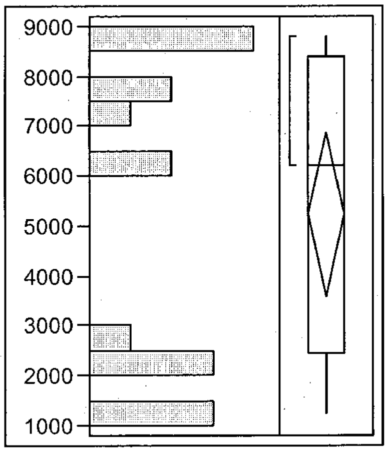
Mean	2590
Std Dev	0
Std Err Mean	0
upper 95% Mean	2590
lower 95% Mean	2590
N	16

Different font
Random

Quartiles

100.0%	maximum	2543.0

Figure A-19, Continued



Choice Right

Quartiles

100.0%	maximum	8800.0
99.5%		8800.0
97.5%		8800.0
90.0%		8781.8
75.0%	quartile	8406.3
50.0%	median	6218.0
25.0%	quartile	2456.8
10.0%		1241.2
2.5%		1223.0
0.5%		1223.0
0.0%	minimum	1223.0

Moments

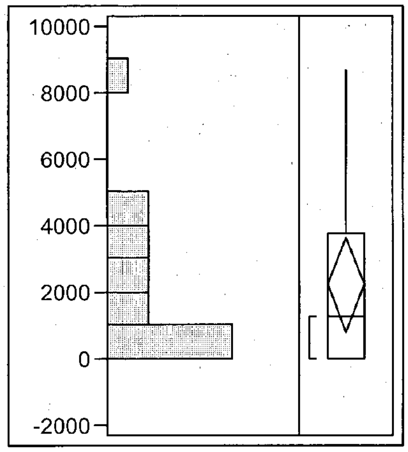
Mean	5223.5
Std Dev	3105.0234
Std Err Mean	776.25585
upper 95% Mean	6878.0502
lower 95% Mean	3568.9498
N	16

99.5%		2543.0
97.5%		2543.0
90.0%		2543.0
75.0%	quartile	2543.0
50.0%	median	2543.0
25.0%	quartile	2543.0
10.0%		2543.0
2.5%		2543.0
0.5%		2543.0
0.0%	minimum	2543.0

Moments

Mean	2543
Std Dev	0
Std Err Mean	0
upper 95% Mean	2543
lower 95% Mean	2543
N	16

Figure A-19, Continued



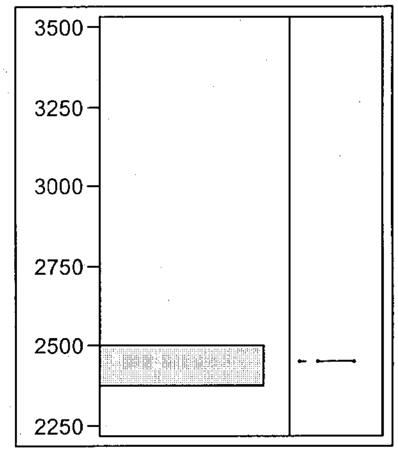
Choice=Stop

Quartiles

100.0%	maximum	8687.0
99.5%		8687.0
97.5%		8687.0
90.0%		6426.2
75.0%	quartile	3748.0
50.0%	median	1243.0
25.0%	quartile	0.0
10.0%		0.0
2.5%		0.0
0.5%		0.0
0.0%	minimum	0.0

Moments

Mean	2216.7333
Std Dev	2569.1329
Std Err Mean	663.34727
upper 95% Mean	3639.4717
lower 95% Mean	793.99495
N	15



Random

Quartiles

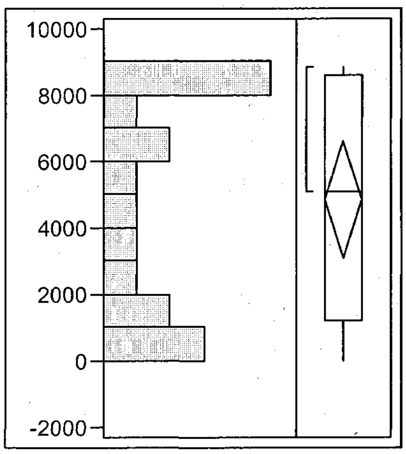
100.0%	maximum	2449.0
99.5%		2449.0
97.5%		2449.0
90.0%		2449.0
75.0%	quartile	2449.0
50.0%	median	2449.0
25.0%	quartile	2449.0
10.0%		2449.0
2.5%		2449.0
0.5%		2449.0
0.0%	minimum	2449.0

Moments

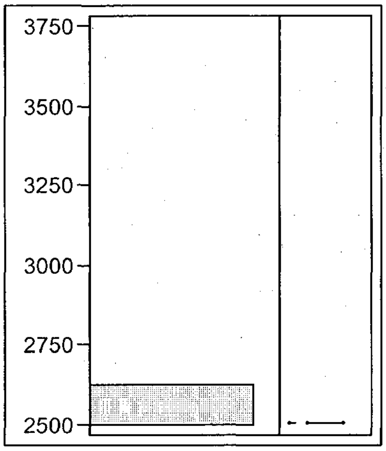
Mean	2449
Std Dev	0
Std Err Mean	0
upper 95% Mean	2449
lower 95% Mean	2449
N	15

Agent Random

Figure A-19, Continued



Choice=Stop



Random

Quartiles (Choice=Stop)

100.0%	maximum	8819.0
99.5%		8819.0
97.5%		8819.0
90.0%		8749.4
75.0%	quartile	8610.0
50.0%	median	5067.0
25.0%	quartile	1207.0
10.0%		0.0
2.5%		0.0
0.5%		0.0
0.0%	minimum	0.0

Moments

Mean	4849.1765
Std Dev	3447.5572
Std Err Mean	836.15545
upper 95% Mean	6621.7468
lower 95% Mean	3076.6061
N	17

Quartiles (Random)

100.0%	maximum	2505.0
99.5%		2505.0
97.5%		2505.0
90.0%		2505.0
75.0%	quartile	2505.0
50.0%	median	2505.0
25.0%	quartile	2505.0
10.0%		2505.0
2.5%		2505.0
0.5%		2505.0
0.0%	minimum	2505.0

Moments

Mean	2505
Std Dev	0
Std Err Mean	0
upper 95% Mean	2505
lower 95% Mean	2505
N	17

Table 36. SME Experimental Results Data

EXPERT TEST RESULTS

Choice	AP Agent Choice (actual)	AP Choice (turn)	Agent Rule Set Choice (turn)	Choice	Sex	Age	Exp.	Mean Agent Random	Mean Expert Choice	Mean Rule Set Choice
L	5105	2590	3118	4	m	69	20	2523	3843	2172
L	0	2590	3118	0	m	63	30	2523	3843	2172
L	6358	2590	3118	5	m	62	10	2523	3843	2172
L	1247	2590	3118	5	m	57	30	2523	3843	2172
L	6401	2590	3118	5	m	53	4	2523	3843	2172
L	0	2590	3118	0	m	60	10	2523	3843	2172
L	0	2590	3118	0	m	51	20	2523	3843	2172
L	4958	2590	3118	4	m	51	20	2523	3843	2172
L	6317	2590	3118	5	m	59	3	2523	3843	2172
L	3733	2590	3118	3	m	59	16	2523	3843	2172
L	0	2590	3118	4	m	69	4	2523	3843	2172
L	0	2590	3118	0	m	68	30	2523	3843	2172
L	0	2590	3118	3	m	52	15	2523	3843	2172
L	7592	2590	3118	6	m	60	24	2523	3843	2172
L	5105	2590	3118	4	m	41	10	2523	3843	2172
L	0	2590	3118	0	m	68	30	2523	3843	2172
R	7516	2543	3389	6	m	69	20	2523	3843	2172
R	2459	2543	3389	2	m	63	30	2523	3843	2172
R	2495	2543	3389	2	m	62	10	2523	3843	2172
R	1223	2543	3389	1	m	57	30	2523	3843	2172
R	1249	2543	3389	1	m	53	4	2523	3843	2172
R	8709	2543	3389	7	m	60	10	2523	3843	2172
R	7453	2543	3389	5	m	51	20	2523	3843	2172
R	2518	2543	3389	2	m	51	20	2523	3843	2172
R	8774	2543	3389	7	m	59	3	2523	3843	2172
R	8800	2543	3389	7	m	59	16	2523	3843	2172
R	8685	2543	3389	7	m	69	4	2523	3843	2172
R	1269	2543	3389	4	m	69	6	2523	3843	2172
R	6218	2543	3389	5	m	52	15	2523	3843	2172
R	2456	2543	3389	3	m	60	24	2523	3843	2172
R	7516	2543	3389	4	m	41	10	2523	3843	2172
R	6218	2543	3389	5	m	68	30	2523	3843	2172
S	2456	2449	564	2	m	69	30	2523	3843	2172
S	0	2449	564	0	m	63	10	2523	3843	2172

Table A-12, Continued

Choice	AP Agent Choice (actual)	AP Choice (turn)	Agent Rule Set Choice (turn)	Choice	Sex	Age	Exp.	Mean Agent Random	Mean Expert Choice	Mean Rule Set Choice
S	3748	2449	564	3	m	62	30	2523	3843	2172
S	3668	2449	564	3	m	57	4	2523	3843	2172
S	4919	2449	564	3	m	53	10	2523	3843	2172
S	4894	2449	564	2	m	51	20	2523	3843	2172
S	0	2449	564	0	m	51	3	2523	3843	2172
S	1180	2449	564	1	m	59	16	2523	3843	2172
S	0	2449	564	0	m	59	4	2523	3843	2172
S	0	2449	564	0	m	69	6	2523	3843	2172
S	8703	2449	564	7	m	69	30	2523	3843	2172
S	0	2449	564	0	m	52	15	2523	3843	2172
S	1243	2449	564	1	m	60	24	2523	3843	2172
S	2456	2449	564	2	m	41	10	2523	3843	2172
S	0	2449	564	0	m	68	30	2523	3843	2172
S	0	2505	564	0	m	69	20	2523	3843	2172
ST	8732	2505	1618	7	m	63	30	2523	3843	2172
ST	7553	2505	1618	6	m	62	10	2523	3843	2172
ST	8819	2505	1618	7	m	57	30	2523	3843	2172
ST	6407	2505	1618	5	m	53	4	2523	3843	2172
ST	6362	2505	1618	5	m	60	10	2523	3843	2172
ST	2479	2505	1618	6	m	51	20	2523	3843	2172
ST	8729	2505	1618	7	m	51	20	2523	3843	2172
ST	3743	2505	1618	3	m	59	3	2523	3843	2172
ST	5067	2505	1618	4	m	59	16	2523	3843	2172
ST	0	2505	1618	0	m	69	4	2523	3843	2172
ST	1227	2505	1618	3	m	69	6	2523	3843	2172
ST	1187	2505	1618	6	m	60	20	2523	3843	2172
ST	8610	2505	1618	7	m	52	15	2523	3843	2172
ST	4911	2505	1618	4	m	60	24	2523	3843	2172
ST	0	2505	1618	4	m	41	10	2523	3843	2172
ST	8610	2505	1618	7	m	68	30	2523	3843	2172

L=left, R=right ,S=stop, ST=straight

Table 37. Random and AP Agent Choice Results

AGENT RANDOM CHOICES

Perceived Choice	AP Agent Choice	Type
L	5105	AP Agent
L	0	AP Agent
L	6358	AP Agent
L	1247	AP Agent
L	6401	AP Agent
L	0	AP Agent
L	0	AP Agent
L	4958	AP Agent
L	6317	AP Agent
L	3733	AP Agent
L	0	AP Agent
L	0	AP Agent
L	0	AP Agent
L	7592	AP Agent
L	5105	AP Agent
L	0	AP Agent
R	7516	AP Agent
R	2459	AP Agent
R	2495	AP Agent
R	1223	AP Agent
R	1249	AP Agent
R	8709	AP Agent
R	7453	AP Agent
R	2518	AP Agent
R	8774	AP Agent
R	8800	AP Agent
R	8685	AP Agent
R	1269	AP Agent
R	6218	AP Agent
R	2456	AP Agent
R	7516	AP Agent
R	6218	AP Agent
S	2456	AP Agent
S	0	AP Agent
S	3748	AP Agent
S	3668	AP Agent
S	4919	AP Agent

Table A-13, Continued

Perceived Choice	AP Agent Choice	Type
S	4894	AP Agent
S	0	AP Agent
S	1180	AP Agent
S	0	AP Agent
S	0	AP Agent
S	8703	AP Agent
S	0	AP Agent
S	1243	AP Agent
S	2456	AP Agent
S	0	AP Agent
S	0	AP Agent
ST	8732	AP Agent
ST	7553	AP Agent
ST	8819	AP Agent
ST	6407	AP Agent
ST	6362	AP Agent
ST	2479	AP Agent
ST	8729	AP Agent
ST	3743	AP Agent
ST	5067	AP Agent
ST	0	AP Agent
ST	1227	AP Agent
ST	1187	AP Agent
ST	8610	AP Agent
ST	4911	AP Agent
ST	0	AP Agent
ST	8610	AP Agent

L=left, R=right, S=stop, ST=straight

B. AGENT SOFTWARE DESCRITPION

Table 384. Software Agent State Space Transition Data

AGENT SPACE TRANSITION DATA

Agent	Transition	**target**	Contacts=>Leaving

Source/target	NewContacts=>
	Trigger
	targetDist > target.zone ||
	D < 0 || vx > v ||
	(xm-target.x)*(xm-target.x)+(ym-target.y)*(ym-target.y) > target.zone*target.zone
	Guard
	Action
	target.setTracked(false);
		Fire	if(perceive.isStateActive(perceive.yperceive) ||
			perceive.isStateActive(perceive.nperceive) ||
			perceive.isStateActive(perceive.Movno) ||
			perceive.isStateActive(perceive.Movyes)
			) {
			if(getTime() - lastevent < 1) //just moved
			return x - 200 + (getTime() - lastevent)*200;
			else return x;
			}
			return x;

Action	if(cues.isStateActive(cues.GestaltCues)) {
	if(getTime() - lastevent < 1) //just moved
	return y - (getTime() - lastevent)*100;
	else return y;
	}
	return y;
		Fire	if(cues.isStateActive(cues.bowave) ||
			cues.isStateActive(cues.bearing) ||
			cues.isStateActive(cues.speed) ||
			cues.isStateActive(cues.distance)
			) {
			if(getTime() - lastevent < 1) //just moved
			return x - 200 + (getTime() - lastevent)*200;
			else return x;
			}
			return x;

Table A-14, Continued

<table>
<tr><td>Action</td><td>if(
approaching_states.isStateActive(
approaching_states.AnticipatedTurn)) {
 if(getTime() - lastevent < 1) //just moved
  return y - (getTime() - lastevent)*100;
 else return y;
}
return y;</td><td>Fire</td><td>if(
approaching_states.isStateActive(
approaching_states.Turn_Right)
||
approaching_states.isStateActive(
approaching_states.Turn_Left) ||
approaching_states.isStateActive(
approaching_states.GestaltCues)
||

approaching_states.isStateActive(
approaching_states.Straight)
) {
 if(getTime() - lastevent < 1) //just moved
  return x - 200 + (getTime() - lastevent)*200;
 else return x;
}
return x;</td></tr>
<tr><td>Fire</td><td>Immediately</td><td>Action</td><td>model.Nright++;</td></tr>
<tr><td>Transition</td><td>Right</td><td>Transition</td><td>Transition</td></tr>
<tr><td>target</td><td>NewContacts=>Straight</td><td>Fire</td><td>Immediately</td></tr>
<tr><td>Fire</td><td>Immediately</td><td>Transition</td><td>transition1</td></tr>
<tr><td>Transition</td><td>Transition</td><td>Source/target</td><td>NewContacts=>Turn Right</td></tr>
<tr><td>Source/target</td><td>NewContacts=>Turn Left</td><td>Fire</td><td>Immediately</td></tr>
<tr><td>Transition</td><td>transition</td><td>target</td><td>Straight=>Leaving</td></tr>
<tr><td>Source/target</td><td>Leaving=>final</td><td>Fire</td><td>If all other guards are closed</td></tr>
<tr><td>Fire</td><td>Immediately</td><td>Action</td><td>model.Nstraight++;</td></tr>
<tr><td>Transition</td><td>Go Straight</td><td>Transition</td><td>Left</td></tr>
<tr><td>target</td><td>Turn Left=>Leaving</td><td>target</td><td>Stop=>Leaving</td></tr>
<tr><td>Fire</td><td>If all other guards are closed</td><td>Fire</td><td>Immediately</td></tr>
<tr><td>Action</td><td>model.Nleft++;</td><td>Guard</td><td>randomTrue(0.1)</td></tr>
<tr><td>Transition</td><td>All Stop</td><td>Action</td><td>model.NStop++;</td></tr>
<tr><td></td><td>Leaving</td><td>action</td><td>//animation
lastevent = getTime();</td></tr>
<tr><td>Entry action</td><td>//animation</td><td>State</td><td>x +=(); y += ()</td></tr>
</table>

Table A-14, Continued

Exit action	astevent = getTime();
	
	x = getX();
		Entry action	Turn Left
			
			//animation

State		**Exit action**	lastevent = getTime(); x += (); y += -()

State	//free Ship
	if(Ship != null) {
	  Ship.free(Ownship.this
	);
	  Ship = null;
	}
	Straight
		State	

Entry action	Turn Right
	
	//animation
	
	lastevent = getTime();
	x += (); y += +();
		Entry action	//free Ship
			if(Ship != null) {
			  Ship.free(Ownship.this);
			  Ship = null;
			}
			NewContacts
			model.NContacts++;

Table 39. HBR AP Xdynamic and Ydynamic Algorithmic Functions

HBR AP GESTALT APPROACHING STATES

Name getX

Type real

Body
```
if( approaching_states.isStateActive( approaching_states.Turn_Right
) ||
   approaching_states.isStateActive( approaching_states.Turn_Left )
||
   approaching_states.isStateActive( approaching_states.Stop ) ||
   approaching_states.isStateActive( approaching_states.Straight )
  ) {
 if( getTime() - lastevent < 1 ) //just moved
   return x - 200 + (getTime() - lastevent)*200;
 else return x;
}
return x;
```

Name getY

Type real

Body
```
if( approaching_states.isStateActive( approaching_states.Leaving ) )
{
 if( getTime() - lastevent < 1 ) //just moved
   return y - (getTime() - lastevent)*100;
 else return y;
}
return y;
```

Variable Own ships

Type Vector

Name free

Type void

Arguments

Type	**Name**
Ownship	cfree

Body
```
if( !Ownships.contains( cfree ) )
 Engine.error( "not contained at free" );
Ownships.remove( cfree );
```

Table A-15, Continued

Name	See code below

```
/handle new NewContact
for( int i=0; i<model.Ownships.size(); i++ ) {
  Ownship c = model.Ownships.item(i);
  if( c.Ship == null && c.approaching_states.isStateActive(
c.approaching_states.NewContacts ) ) {
    c.Ship = Ship.this;
    c.setModified();
    Ownships.add( c );
    break;
  }
}
```

Table 40. HBR AP Xdynamic and Ydynamic Algorithmic Functions

HBR AP GESTALT CUE DERIVATIVES

Startup code

```
_ref_tx .connect( target._ref_x  );
_ref_ty .connect( target._ref_y  );
_ref_tvx.connect( target._ref_vx );
_ref_tvy.connect( target._ref_vy );

target = (Target)getOwner();
x = target.x;
y = target.y;
```

ExpertNoviceArea

Equations

```
tm = (tx-x)/(vx-tvx)
D = b*b-4*a*c
c = sqr(tvy*(tx-x)-tvx*(ty-y))-sqr(tx-x)*v*v
b = 2*(ty-y)*(tvy*(tx-x)-tvx*(ty-y))
a = sqr(tx-x)+sqr(ty-y)
vx = (-b-sqrt(abs(D)))/(2*a)
vy = sqrt(abs(v*v-vx*vx))*sign(ty+tvy*tm-y)
d(x)/dt = vx
d(y)/dt = vy
```

ExpertNoviceArea

```
targetDist = sqrt( sqr(x-radar.x) + sqr(y-target.y) )

targetDist = sqrt( a )
xm = x+vx*tm
ym = y+vy*tm
```

Table A-16 Continued

ExpertNovice Area

Additional class code

```
public ExpertNoviceArea( shipcontact target ) {
  this.target = target;
}

ExpertNoviceArea() {
}

private void disconnect() {
        _ref_tx .disconnect( target._ref_x  );
        _ref_ty .disconnect( target._ref_y  );
        _ref_tvx.disconnect( target._ref_vx );
        _ref_tvy.disconnect( target._ref_vy );
}

private static double sqr( double x ) {
  return x*x;
}

private static double sign( double x ) {
  return x==0?0:x>0?1:-1;
}

shipcontact target;
```

Own ships

```
AP = (AP)getOwner();
//animation
x = radius*cos( angle );
y = radius*sin( angle );
```

Function Cues

Get x,y

```
if( cues.isStateActive( cues.bowave ) ||
    cues.isStateActive( cues.bearing ) ||
    cues.isStateActive( cues.speed ) ||
    cues.isStateActive( cues.distance )
   ) {
  if( getTime() - lastevent < 1 ) //just moved
    return x - 200 + (getTime() - lastevent)*200;
  else return x;
}
return x;
```

Table A-16, Continued

	Argument
Contact velocity	

```
int ExpertNoviceAreaIndex = 0;

public void perceiveExpertNoviceArea( shipcontact e )
{
 ExpertNoviceArea m = new ExpertNoviceArea( e );
        m.set_v( ExpertNoviceAreaVelocity );
        /*
        if ( m.v * 0.99 < e.v ) {
         // make it happen
              return;
        }*/
 setup_ExpertNoviceAreas( m,
ExpertNoviceAreaIndex++ );
 e.setTracked( true );
}
```

Cue Derivative

```
d(distance)/dt = distance_cue
d(bowave)/dt = bowave_cue
d(speed)/dt = speed_cue
d(bearing)/dt = bearing_cue
Additional class code
int shipcontactIndex = 0;
public void removeshipcontact( shipcontact obj ) {
 dispose_shipcontacts( obj );
}
```

Cue Gestalt

```
if( cues.isStateActive( cues.GestaltCues ) ) {
 if( getTime() - lastevent < 1 ) //just moved
  return y - (getTime() - lastevent)*100;
 else return y;
}
return y;
```

Target Tug Vector

```
public boolean isAlive() {
 return main.getCurrentState() == main.Normal;
}

public boolean isInRange() {
 return main.getCurrentState() == main.InRange;
```

Table A-16, Continued

```
}
```

Target Tug Vector

```
public void onCreate() {
        x = DistrUniform.sample( 470, 0 )*.3;
        y = DistrUniform.sample( 130, 0 );
        x = DistrUniform.sample( 500, 0 )*.7;
        y = DistrUniform.sample( 210, 0 );
        x = DistrUniform.sample( 600, 0 )*.1;
        y = DistrUniform.sample( 310, 0 );
        x = DistrUniform.sample( 700, 0 );
        y = DistrUniform.sample( 410, 0 );
}
```

Ship Contact

```
d(x)/dt = vx
d(y)/dt = vy
dist = sqrt( (tx-x)*(tx-x) + (ty-y)*(ty-y) )
Additional class code
Tugtargetvector target;
boolean tracked = false;
public boolean isDead = false;

public void onCreate() {
        x  = 0;
        y  = DistrUniform.sample(  230, 230 );
        target = chooseTarget();
 // if there is no one alive target - select random
 if (target == null) target = chooseRandomTarget();
        updateVelocity();
}

private Tugtargetvector chooseTarget() {
        // choose random target
        Root root = (Root)Engine.getRoot();
        Enumeration e =
root.Tugtargetvectors.elements();
        int alive = 0;
        while ( e.hasMoreElements() ) {
         if (
((Tugtargetvector)e.nextElement()).isAlive() )
           alive++;
        }
        if ( alive > 0 ) {
         int t = (int)DistrUniform.sample( alive );
         e = root.Tugtargetvectors.elements();
         while ( e.hasMoreElements() ) {
```

Table A-16, Continued

Ship Contact

```
                        Tugtargetvector b =
(Tugtargetvector)e.nextElement();
                        if ( b.isAlive() && --t < 0 ) {
                                    // this will be the target
                                    return b;
                              }
                        }
            }
            return null;
}


private Tugtargetvector chooseRandomTarget() {
            // choose random target
            Root root = (Root)Engine.getRoot();

  int totalTugtargetvectors =
root.Tugtargetvectors.size();

  int t = (int)DistrUniform.sample(
totalTugtargetvectors - 1 );

  return root.Tugtargetvectors.item( t );

}

private void updateVelocity() {
            if ( target == null ) {
              // self destroy
                        destroyed.receive( new Object() );
                        return;
            }
  tx = target.x;
            ty = target.y;
            vx = tx-x;
            vy = ty-y;
  double v1 = Math.sqrt( vx*vx + vy*vy );
            vx *= v/v1;
            vy *= v/v1;
            if ( vx < 0 ) {
              vx *= -1;
                        vy *= -1;
            }
}

public void setTracked( boolean t ) {
```

Table A-16, Continued

Ship Contact

```
 tracked = t;
}

public boolean isTracked() {
 return tracked;
}

public boolean isInRange() {
 return main != null ?
          main.getCurrentState() == main.InRange :
false;
}
```

Table 41. AnyLogic™ Simulation Functions

SIMULATION FUNCTIONS

Return type	*Name*	*Description*
void	applyDefaultLayout()	Re-applies the default layout settings to the current population.
double	getX()	Returns the current agent x-coordinate; takes into account movement. CONTINUOUS space only.
double	getY()	Returns the current agent y-coordinate; takes into account movement. CONTINUOUS space only.
void	moveTo (double x, double y)	Starts moving the agent from the current location to (x,y) with the current Velocity. Stops the previous movement, if any. CONTINUOUS space only.
void	jumpTo (double x, double y)	Immediately puts the agent to the location (x,y). Stops the previous movement, if any. CONTINUOUS space only.
void	stop()	Stops the previous movement, if any, leaving the agent at the current position. CONTINUOUS space only.

Table A-17, Continued

Return type	*Name*	*Description*
double	timeToArrival()	Returns the remaining time to arrival if the agent is currently moving as a result of calling moveTo(), 0 otherwise. CONTINUOUS space only.
double	distanceTo (ActiveObject other)	Returns the distance to another agent of the same population. CONTINUOUS space only.
int	getC()	Returns the column of cell occupied by the agent. DISCRETE space only.
int	getR()	Returns the row of cell occupied by the agent. DISCRETE space only.
int[]	findRandomEmptyCell()	Returns the array [row,column] of a random empty cell; throws exception if there are no empty cells. DISCRETE space only.
void	jumpToCell(int r, int c)	Puts the agent into a cell [r,c]; throws exception id this cell is already occupied. DISCRETE space only.
void	jumpToRandomCell()	Puts the agent into a random empty cell; throws exception if there are no empty cells. DISCRETE space only.
ActiveObject	getAgentAtCell (int r, int c)	Returns agent at the specified cell; null if the cell is empty of indices outside their ranges. DISCRETE space only.

Table A-17, Continued

Return type	*Name*	*Description*
Vector	getNeighbors()	Returns the Vector containing the neighboring agents, depending on the current neighborhood model. DISCRETE space only.
ActiveObject	getAgentN()	Return the age corresponding direction from the cell occupied by this agent, null if the cell is empty or outside range. DISCRETE space only.
ActiveObject	getAgentNE()	
ActiveObject	getAgentE()	
ActiveObject	getAgentSE()	
ActiveObject	getAgentS()	
ActiveObject	getAgentSW()	
ActiveObject	getAgentW()	
ActiveObject	getAgentNW()	
void	applyDefaultNetwork()	Re-applies the default network settings to the current population.
Vector	getContacts()	Returns the Vector of agents that are connected to this agent. Network type cannot be NONE.
void	connectTo (ActiveObject other)	Connects this agent to the other agent of the same population. Network type cannot be NONE.

Table A-17, Continued

Return type	*Name*	*Description*
void	disconnectFrom (ActiveObject other)	Disconnects this agent to the other agent of the same population. Network type can not be NONE.
void	disconnectFromAll()	Disconnects this agent from all other agents in the same population. Network type can not be NONE.
void	sendTo(Object msg, ActiveObejct agent)	Sends msg to the agent of the same population.
void	sendToAll (Object msg)	Broadcasts msg to all agents of the same population (including this one).
void	sendToRandom (Object msg)	Sends msg to a random agent of the same population (this object may be chosen as well).
void	sendToAllContacts (Object msg)	Broadcasts msg to all agents connected to this agent. Network type cannot be NONI
void	sendToRandomContact (Object msg)	Sends msg to a random agent connected to this agent. Network type cannot be NONI
code	OnBeforeStepGlobal	Action that is executed once for the whole population at the beginning of a step before OnBeforeStep. DISCRETE time only.
code	OnBeforeStep	Action that is executed within a step for each agent after OnBeforeStepGlobal and before OnStepGlobal. DISCRETE time only.

Table A-17, Continued

Return type	*Name*	*Description*
code	OnStepGlobal	Action that is executed within a step once for the whole population after OnBeforeStep and before OnStep. DISCRETE time only.
code	OnStep	Action that is executed at the end of a step for each agent after OnStepGlobal. DISCRETE time only.
code	OnReceive	The action called upon receiving a message from another agent. Use message to retrieve the message and sender to retrieve the sending agent.
double	Velocity	10 In STATIC OR MOBILE location control mode: the velocity of the agent when it moves if you call moveTo. Can be changed dynamically at any time.
code	OnArrival	The code executed when the agent arrives to the destination specified in the call of moveTc
code<double>	Xdynamic	In DYNAMIC USER DEFINED location control mode: the code to obtain the current coordinates of the agent.
code	OnClick	The code executed when the mouse is clicked on the agent animation.

Table A-17, Continued

Return type	*Name*		*Description*
boolean	ShowInfoStringOnClick		Allows toggling of the InfoString on click.
double	ContactsPerAgent		Parameter for RANDOM, RING LATTICE and SMALL WORLD network types
double	ContactRange		Parameter for ALL IN RANGE network type.
boolean	ContactLine Semitransparent		Transparency of contact lines (note that displaying a large number of semitransparent lines may significantly slow down the model).
code	OnReceive		The action called upon receiving a message from another agent. Use message to retrieve the message and sender to retrieve the sending agent.
int	SpaceRows		The number of cell rows and columns in DISCRETE space
int	Rinitial		In DISCRETE space: the initial row and column of the agent's cell.
int	Space	CONTINUOUS	Space mode: CONTINUOUS or DISCRETE.
int	DefaultLayout Contiouous	RANDOM	In CONTINUOUS space: the default layout that is applied at the startup.

Table 17, Continued

Return type	*Name*		*Description*
int	DefaultLayo utDiscrete	RANDOM	In DISCRETE space: the default layout that is applied at the startup.
double	SpaceWidth	500	In CONTINUOUS space: the space dimensions that are used while the default layout is being applied.
double	SpaceHeight	500	
int	Location Continuous	STATIC OR MOBILE	The way the user wants to cor location of the agent in CONT space
double	Xinitial	RANDOM	In STATIC OR MOBILE location control mode: the initial coordinates of the agent.
double	Yinitial		RANDOM
code<double>	Ydynamic		
int	SpaceColumns		
double	CellWidth		Cell dimension on the animation.
int	Cinitial		
boolean	ShowInfoStringOnClick		Allows toggling of the InfoString on click.
code<String>	InfoString	"Agent #" + getOwner().getIn dex()	The dynamic textual information that may be displayed on click.
Color	InfoStringColor	Color.blue	The color of the textual information.
int	DefaultNetwork	NONE	The network type that is applied at the startup
double	NeighborLinkPr obability	0.95	Parameter for SMALL WORLD network type.

Table A-17, Continued

Return type	*Name*		*Description*
double	ScaleFreeM	10	Parameter for SCALE FREE network type.
int	ShowContacts	ON CLICK	Defines the display mode of contact lines.
Color	ContactLine Color	Color.black	Color of contact lines.
int	Neighborhood	EUCLIDIAN	Defines what cells are considered as neighboring.

Figure A-20. Parameter Variation Test for Random HBR AP model

PARAMETER VARIATION TEST FOR EXPERT SYSTEM

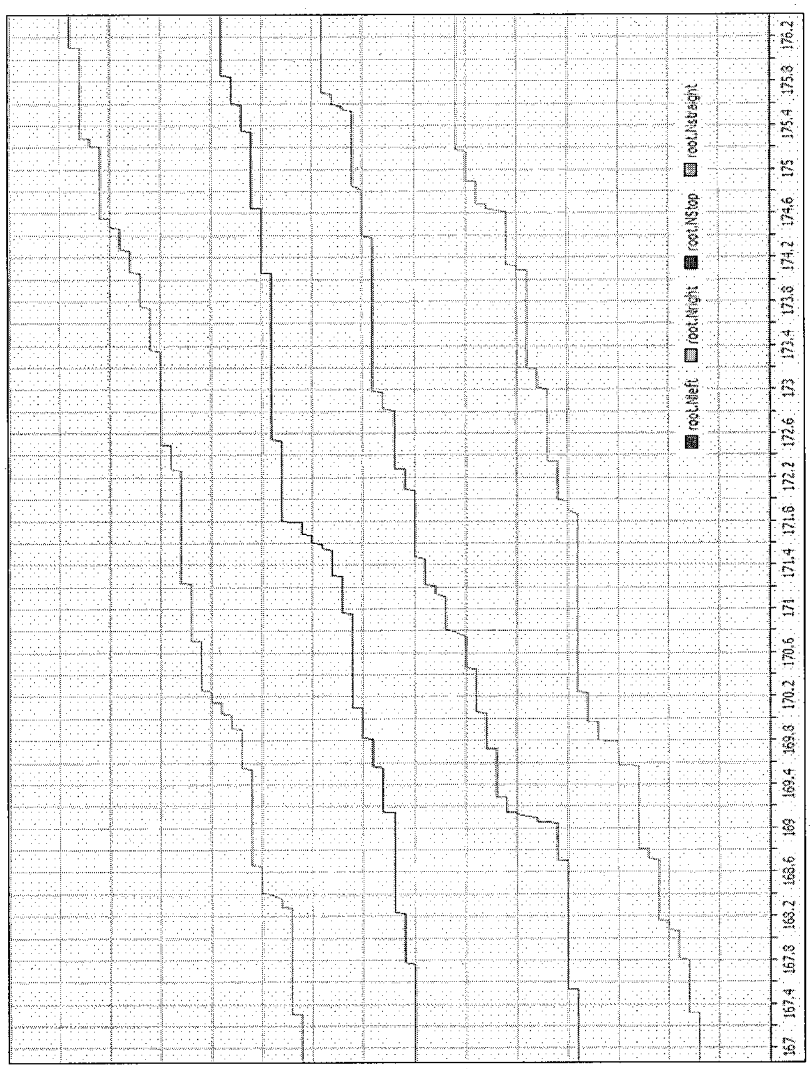


Figure A-21. Parameter Variation Test for HBR AP model

PARAMETER VARIATION TEST FOR AP MODEL

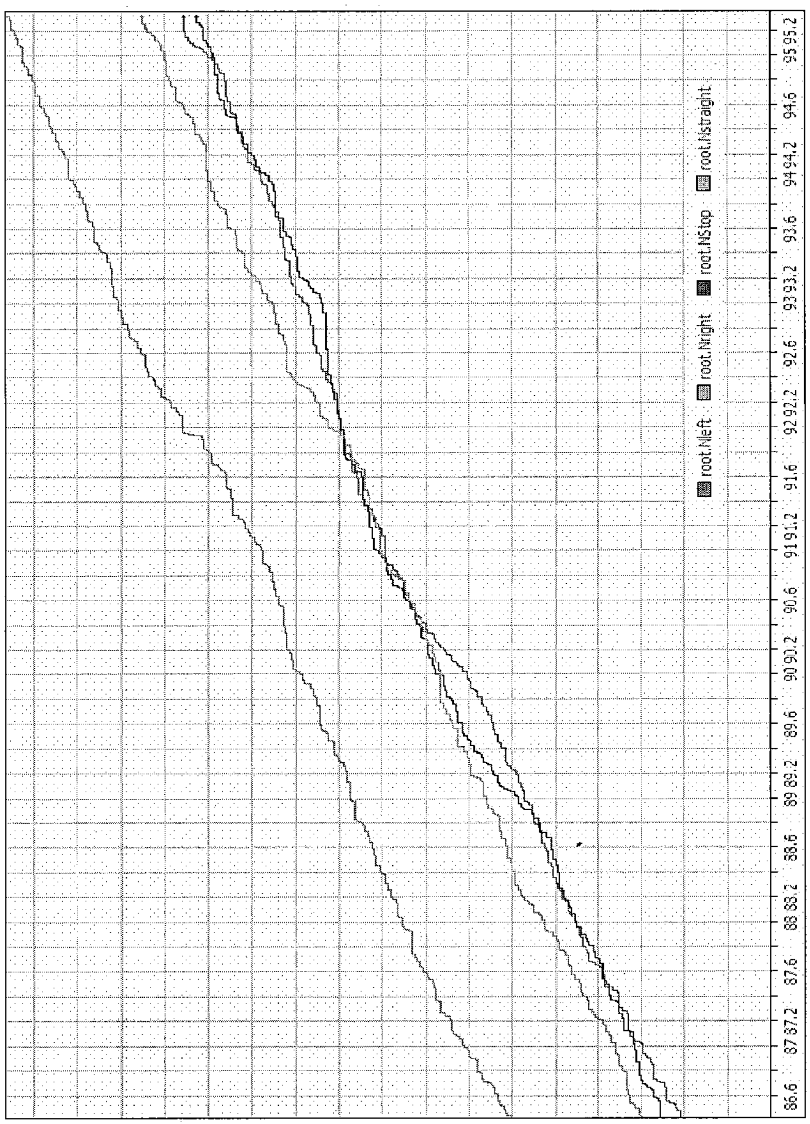
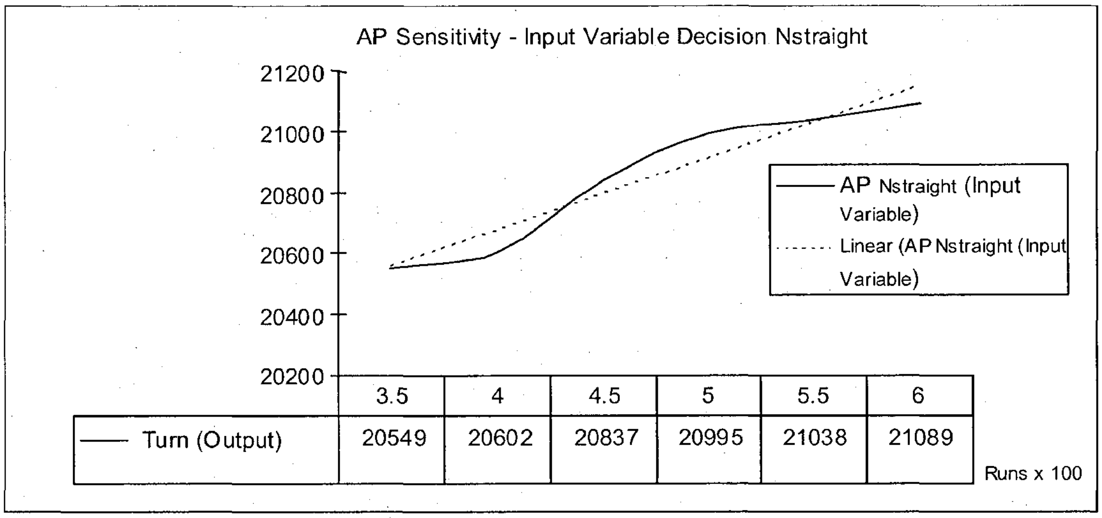


Figure A-22. Post-Trial Parameter Variation Test Data for Human Nstraight Decision

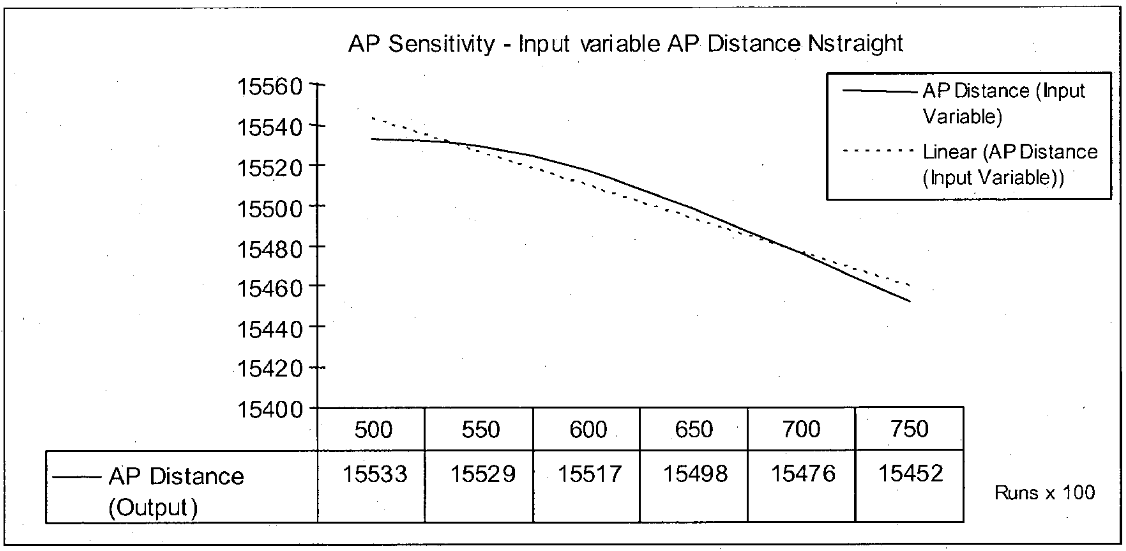
SENSITIVITY TEST FOR N-STRAIGHT DECISION (EXPERT)



AP Nstraight (Input Variable)	Left	Right	Stop	Straight
3.5	5166	5150	5135	20549
4	5122	5132	5144	20602
4.5	5072	5061	5030	20837
5	5053	5054	4897	20995
5.5	5039	5037	4886	21038
6	4985	5030	4896	21089
Avg	5073.06	5077.40	4998.08	**20851.76**

Figure A-23. Post-Trial Parameter Variation Test Data for AP Nstraight Distance

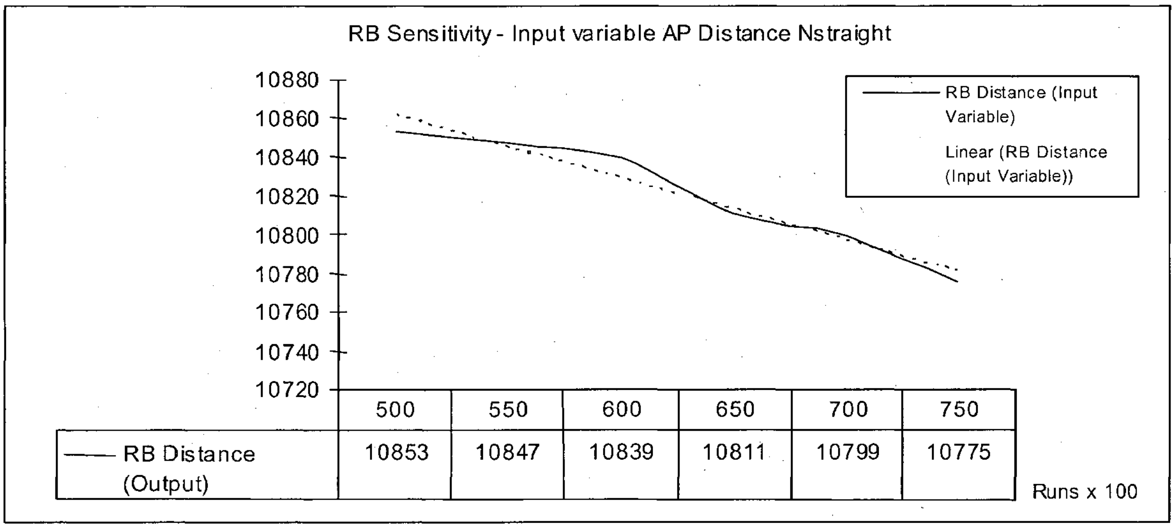
SENSITIVITY TEST FOR N-STRAIGHT DECISION (AP AGENT)



AP Distance (Input Variable)	Left	Right	Stop	Straight
500	6905	11220	2342	15533
550	6919	11227	2325	15529
600	6923	11242	2318	15517
650	6953	11261	2288	15498
700	6972	11285	2267	15476
750	7001	11303	2244	15452
Avg	6945.50	11256.33	2297.33	**15500.83**

Figure A-24. Post-Trial Parameter Variation Test Data for Rule-Based N-Straight Distance

SENSITIVITY TEST FOR N-STRAIGHT DECISION (RULE-BASED)



RB Distance (Input Variable)	Left	Right	Stop	Straight
500	6536	10012	8599	10853
550	6539	10017	8597	10847
600	6549	10023	8589	10839
650	6571	10041	8577	10811
700	6586	10044	8571	10799
750	6599	10065	8561	10775
Avg	6563.33	10033.67	8582.33	**10820.67**

C. ODU IRB INFORMED CONSENT

INTERNAL REVIEW BOARD (IRB)

INFORMED CONSENT DOCUMENT

(OLD DOMINION UNIVERSITY)

31 March 2006

PROJECT TITLE: Validation of Human Behavior Representation (HBR) Artificial Perception in a Simulated Environment

INTRODUCTION:

The purposes of this form are to give you information that may affect your decision whether to say YES or NO to participation in this research, and to record the consent of those who say YES. This research is designed to validate artificial perception in a simulated environment and is partial fulfillment of a Ph.D. Dissertation.

RESEARCHERS

Principal Research Investigator (RPI):

Ryland C. Gaskins III, Ph.D.
Senior Research Scientist
Virginia Modeling and Simulation Center (VMASC)
Department of Engineering and Technology
College: College of Engineering
Department of Engineering and Technology

Primary Investigator:

Randall B. Garrett, Ph.D. Candidate
Virginia Modeling and Simulation Center (VMASC)
Department of Engineering and Technology
College: College of Engineering
Department of Engineering and Technology

DESCRIPTION OF RESEARCH STUDY:

Several studies have been conducted looking into the subject of validating representations of human actions in computer simulations. None of them have explained how close to reality human actions are with those represented by computer "agent-objects' in a simulated environment that represents the same real life scenario. The purpose of this research is to survey experienced ship navigators and perform a test to see if a "navigator" simulated agent will respond similar to the data given by the participants.

If you decide to participate, then you will join a study involving research of critical navigation scenarios. Participants are asked to:

Identify several ship critical harbor/channel navigation scenarios that require accurate decision making for ensuring safe navigation.

Review a computer simulation of the critical navigation scenario to ensure that it is an accurate "realistic" representation.

Participate in a computer generated interactive simulation test of the navigation scenario. The simulation will be stopped at certain intervals and the participant will be asked to score from 0 (least likely) to 9 (most likely) that anticipated events will or will not occur. These events will be given prior to the simulation.

EXCLUSIONARY CRITERIA:

You should have completed at least 5 years in a position that required you to frequently navigate channels, harbors or waterways whose navigation may have proven potentially hazardous. To the best of your knowledge, you should not have physical conditions, such as carpal tunnel, vision problems or health issues that may be aggravated through normal interaction with a computer or through simulation that would keep you from participating in this study.

RISKS AND BENEFITS:

RISKS: If you decide to participate in this study, then you may face risks similar to those experienced through the normal use of a desk or laptop computer. If you have carpal tunnel syndrome, visual impairments or any health risks associated with normal computer use, then you should not consent to this experiment. The researcher tried to reduce these risks by limiting the exposure to a single instance or scenario displayed on a computer screen. In addition, as with any research, there is some possibility that you may be subject to risks that have not yet been identified.

BENEFITS: The main benefit to you for participating in this study is that you may contributing to the reduction of ship navigation accidents through the recognition of faulty reality and subsequent risk related decisions.

COSTS AND PAYMENTS:

The researchers want your decision about participating in this study to be voluntary. Yet they recognize that your participation may pose some inconvenience such as time constraints and schedule changes. The researchers are unable to give you any payment for participating in this study.

NEW INFORMATION:

If the researchers find new information during a study that would reasonably change your decision about participating, then they will give it to you.

CONFIDENTIALITY:

All information obtained about you in this study is strictly confidential unless disclosure is required by law. The results of this study may be used in reports, presentations and publications, but the researcher will not identify you.

WITHDRAWAL PRIVILEGE:

It is OK for you to say NO. Even if you say YES now, you are free to say NO later, and walk away or withdraw from the study -- at any time. Your decision will not affect your relationship with Old Dominion University, the researchers, or otherwise cause a loss of benefits to which you might otherwise be entitled

COMPENSATION FOR ILLNESS AND INJURY:

If you say YES, then your consent in this document does not waive any of your legal rights. However, in the event of harm, injury, or illness arising from this study, neither Old Dominion University nor the researchers are able to give you any money, insurance coverage, free medical care, or any other compensation for such injury. In the event that you suffer injury as a result of participation in this research project, you may contact Dr. Ryland C. Gaskins III, the Principal Research Investigator, telephone 757-686-6233, at Old Dominion University, who will be glad to review the matter with you.

VOLUNTARY CONSENT:

By signing this form, you are saying several things. You are saying that you have read this form or have had it read to you, that you are satisfied that you understand this form, the research study, and its risks and benefits. The researchers should have answered any questions you may have had about the research. If you have any questions later on, then the researchers should be able to answer them:

If at any time you feel pressured to participate, or if you have any questions about your rights or this form, then you should Dr. Ryland C. Gaskins III, The Principal Research Investigator, telephone 757-686-6233, or the Old Dominion University Office of Research, at 757-683-3460.

In addition, importantly, by signing below, you are telling the researcher YES, that you agree to participate in this study. The researcher should give you a copy of this form for your records.

INVESTIGATORS' STATEMENT:

I certify that I have explained to this subject the nature and purpose of this research, including benefits, risks, costs, and any experimental procedures. I have described the rights and protections afforded to human subjects and have done nothing to pressure, coerce, or falsely entice this subject into participating. I am aware of my obligations under state and federal laws, and promise compliance. I have answered the subject's questions and have encouraged him/her to ask additional questions at any time during the course of this study. I have witnessed the above signature(s) on this consent form.

VITA

Randall Bartholomew Garrett
Old Dominion University

4609 Grove Hill Court
Virginia Beach, Virginia 23456

Education:

Ph.D. Modeling and Simulation, Virginia Modeling, Analysis and Simulation Center (VMASC), College of Technology and Engineering, Old Dominion University, Norfolk, Virginia

M.S., Safety and Technology Management, College of Information Technology and Management, Marshall University, Huntington, WV. (1986)

B.A., Sociology and Economics, University of Arkansas, Fayetteville, AR. (1976)

Professional History:

2004-Present, General Dynamics Advanced Information Systems (GDAIS), Suffolk, VA; Technical lead, program and contract manager for GDAIS Modeling and Simulation (M&S) programs

2000-2004, Anteon Systems Engineering Group (SEG), Virginia Beach, VA; IT Manager for Anteon SEG, Virginia Beach

1998-2000, Systems Engineer/Software/ Design, LAN/WAN Manager, EDO Corporation, Chesapeake, VA; Senior Engineer for EDO Corporation

1994-1998, Research and Development/ Engineer/Chief Operating Officer, SRT Corporation, Virginia Beach, VA; Information systems and software development manager; Chief Operations Officer for Research and Development (R&D); Member Board of Directors

1994 - Naval Officer, United States Navy (Retired)

Professional Affiliations:

American Association for the Advancement of Science (AAAS)
Human Factors and Ergonomics Society
IEEE
Rotary Club of Hampton Roads
Armed Forces Communications Electronics Association (AFCEA)

Awards and Honors:

Three Chief of Naval Operations "Certificates of Merit"
Department of Defense service awards/medals
West Virginia State "Spirit of Freedom" award, - presented by the Secretary of State
"Honorary Huntitonian" award - presented by the Huntington Mayor
Rotary International "Paul Harris Fellow"